



HAL
open science

Leveraging an Isolation Forest to Anomaly Detection and Data Clustering

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

► **To cite this version:**

Véronne Yepmo, Grégory Smits, Marie-Jeanne Lesot, Olivier Pivert. Leveraging an Isolation Forest to Anomaly Detection and Data Clustering. *Data and Knowledge Engineering*, 2024, 151, pp.102302. 10.1016/j.datak.2024.102302 . hal-04516593

HAL Id: hal-04516593

<https://hal.science/hal-04516593v1>

Submitted on 22 Mar 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Leveraging an Isolation Forest to Anomaly Detection and Data Clustering

Véronne Yepmo^{a,*}, Grégory Smits^b, Marie-Jeanne Lesot^c, Olivier Pivert^a

^a*Université de Rennes, IRISA, Lannion, France*

^b*IMT Atlantique, Lab STICC, Brest, France*

^c*Sorbonne Université, LIP6, Paris, France*

Abstract

Understanding why some points in a data set are considered as anomalies cannot be done without taking into account the structure of the regular points. Whereas many machine learning methods are dedicated to the identification of anomalies on one side, or to the identification of the data inner-structure on the other side, a solution is introduced to answers these two tasks using a same data model, a variant of an isolation forest. The initial algorithm to construct an isolation forest is indeed revisited to preserve the data inner structure without affecting the efficiency of the outlier detection. Experiments conducted both on synthetic and real-world data sets show that, in addition to improving the detection of abnormal data points, the proposed variant of isolation forest allows for a reconstruction of the subspaces of high density. Therefore, the former can serve as a basis for a unified approach to detect global and local anomalies, which is a necessary condition to then provide users with informative descriptions of the data.

Keywords: Anomaly/Outlier detection, Isolation Forest, Clustering

1. Introduction

Because of its numerous applications ranging from spam detection [1] to cancer detection [2], anomaly detection has been extensively studied and now constitutes a research field in itself [3]. The Isolation Forest (IF) [4] is one of the most appealing anomaly detection methods. This is due to the fact that it is unsupervised, fast and has few hyper-parameters. An IF is a set of binary trees, each of them is constructed by recursively partitioning the data space at random, with the aim to isolate points into its leaves. As many outlier detection methods, IF returns a subset of points identified as anomalies because of their

*Corresponding author

Email addresses: `veronne.yepmo-tchaghe@irisa.fr` (Véronne Yepmo), `gregory.smits@imt-atlantique.fr` (Grégory Smits), `marie-jeanne.lesot@lip6.fr` (Marie-Jeanne Lesot), `olivier.pivert@irisa.fr` (Olivier Pivert)

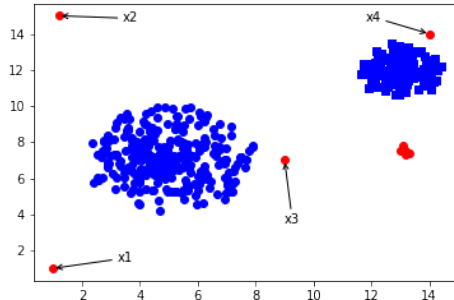


Figure 1: 2D toy data set illustrating the notions of global and local anomalies

10 high anomaly score. However, in many applicative contexts, and so as to make
 the right corporate decision, end users often need to know the reasons why a
 data point is considered as an anomaly. Explaining anomalies has therefore
 become a crucial issue that has received some attention during the last decade
 among the machine learning community [5, 6]. It has been pointed out in [7]
 15 that understanding the provenance of found anomalies ideally relies on a con-
 trastive comparison with the structural properties of so-called regular points.
 The identified anomalies are therefore explained in relation to one or several
 groups of regular data, and not as isolated points from the rest of the data. As
 an illustrative example, Figure 1 depicts, on a toy 2-dimensional dataset, the
 20 difference between global anomalies (x_1 and x_2 for instance) and local anom-
 alies: points x_3 and x_4 may indeed be considered respectively as deviations from
 the cluster of circles and the cluster of squares. Although the existence of local
 anomalies is acknowledged in the literature, that local context is often forgotten
 during the explanation: anomalies are generally explained as if they deviate
 25 from all the other instances in the data set. An explanation for the abnormality
 of x_3 would therefore be that its value for feature f_1 (x -axis) is too high for
 instance, which is an incomplete explanation in this context. A complete expla-
 nation for x_3 would be that it seems to belong to the cluster of circles, because
 of its value on feature f_2 (y -axis), but deviates from that cluster because the
 30 value on f_1 is too high. However, only few of the existing approaches reach this
 level of detail in the provided explanations. This paper takes a step towards the
 extraction of such contrastive explanations between anomalies and the intrinsic
 structure of regular points.

The goal of this work is to provide a unified solution to both the detection
 35 of anomalies and the identification of the regular points inner structure. Having
 such structural knowledge available is a prerequisite for explaining the reason
 why some points are considered as outliers. The contribution of this paper is
 therefore to consider an IF as a unified data model that can be used to iden-
 tify isolated points and dense regions of regular points as well. By reconsidering
 40 random separations that split dense regions of points, leaves of the obtained iso-

lation trees may contain isolated points, corresponding to anomalies, or groups of similar points.

The objective of the method proposed in this paper, named RIFIFI that stands for *Revised Isolation Forest to Identify Fraud¹ and the data Inner structure*, is to preserve as much as possible the structural information during the construction of the isolation forest, in order to reconstruct the regular point clusters. To do so, the selection process of the separations is revisited, without changing its complexity: it is no longer completely random, but guided by the will to preserve as much as possible the proximity between the points belonging to the same data cluster. This contribution is a first step toward a complete data-structure-aware anomaly explanation strategy.

The remainder of the paper is organized as follows. Section 2 positions the proposed approach in relation to existing works in the field of anomaly detection and explanation, especially in connection with the data inner structure. The initial algorithm of IF is recalled in this section as well. The proposed RIFIFI method is described in Section 3. Then, experiments showing the relevance of the approach to build a partition of regular data are presented in Section 4. Some perspectives of this work are finally discussed in Section 5.

2. Related Work

This section positions the proposed approach against existing works in the field of anomaly detection, anomaly explanation and clustering with outlier management capabilities.

2.1. Anomaly Detection

Anomaly detection in machine learning can be considered as a supervised, semi-supervised or unsupervised problem. The supervised and semi-supervised settings both require labels and rely on two successive steps: training and testing. In the supervised setting, the training step is performed on regular and abnormal data, whereas in the semi-supervised setting, only regular instances are used during training. The unsupervised case is the most attractive one because of the unpredictability of anomalies and the difficulty of labeling data sets. Local Outlier Factor (LOF) [8], One-Class Support Vector Machines [9] and IF [4] are among the most popular unsupervised methods even if many other approaches exist, e.g. using autoencoders [10], correlation laws [11], etc. The contribution of this work being an extended use of an isolation forest, a focus is given on this particular anomaly detection method.

IF is an algorithm dedicated to the detection of global anomalies relying on the principle that an anomaly may be easily separated from the rest of the data. It is an ensemble-based algorithm as a forest is composed of t trees, each

¹The term fraud is used in the name as a synonym of anomaly because this work takes place in a large research project aiming at detecting suspicious goods shipped across seas.

Algorithm 1 Isolation Forest : *build_tree* [4]

Inputs: a sample $D \subset \mathcal{D}$, the depth d of the current node
Output: a node in an isolation tree
if $|D| = 1$ or $d > h_{lim}$ **then**
 Return $node(null, null, D, d, null, null)$ \triangleright Leaf (terminal node)
else
 $A \leftarrow random(\mathcal{A})$ \triangleright Random attribute selection
 $v \leftarrow random(range(A))$ \triangleright Random value selection
 $D_l \leftarrow \{x \in D/x.A < v\}$
 $D_r \leftarrow \{x \in D/x.A \geq v\}$
 Return $node(build_tree(D_l, d + 1),$ \triangleright Internal node
 $build_tree(D_r, d + 1), D, d, A, v)$
end if

of them contributing in the calculation of an anomaly score attached to each
80 data point.

Each isolation tree of an IF is built on a randomly drawn sample D of the
dataset \mathcal{D} (Table 1 recaps the notations used throughout the paper). At each
step of the construction of the tree (see Algorithm 1), an attribute A and a
value v in the range of values observed for A in D are selected randomly. The
85 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. The process is repeated
recursively from the root of the tree that contains all the sample data, until one
of the following two conditions is met:

- the node is no longer separable (it contains a single point) ;
- 90 • the depth limit of a tree, 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
95 and RN are pointers to its left and right node respectively, $D \subseteq \mathcal{D}$, $d \in \mathbb{N}$ is
its depth in the tree, $A \in \mathcal{A}$ and $v \in dom(A)$. 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
100 the node containing the data point in each tree, is computed as follows [4]:

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

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.

Several variants of isolation forests 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 [12] where five new functions to compute anomaly scores are proposed. Others modify the construction of the trees but not the calculation of scores. In [13] and [14], oblique separations are used, but with different goals: detecting clusters of anomalies for the former and improving score consistency for the latter. In [15], the separations are no longer completely random and aim at minimizing the weighted standard deviation of the subtrees depth induced by each separation. The binary separation operated in the initial IF approach has been reconsidered in [16] to take into account the local data inner structure. Instead of separating a set of points into two child nodes, a k -means algorithm combined with an elbow rule to determine the value of k is applied at each step of the tree construction. Each found cluster forms a leaf and the anomaly score now depends on the distance of the point with the limits of the cluster it is assigned to. This extension of the IF strategy, in addition to introducing a prohibitive cost overhead during tree construction, relies on the definition of a distance metrics and do not aim at reconstructing the global data inner structure as clusters are only very locally identified, i.e. at the node level.

The RIFIFI method proposed in this paper also produces not completely random separations with the objective of both efficiently isolate outliers and preserve the regular data points structure. In a sense, RIFIFI takes the opposite direction of the extension of random forests proposed in [17] where a split is randomly selected among k candidates that maximise a node splitting function. RIFIFI reconsiders according to a density criterion a random node splitting function.

2.2. Anomaly Explanation

Anomaly explanation has received less attention than classification explanation. Yet, because of the diverse nature of anomalies, anomaly explanation deserves special treatment even though it has benefited from works dedicated to the explanation of classifiers and neural networks outputs. In [7], four categories of explanations have been identified: attribute importance explanation, attribute value explanation, point comparison explanation, and intrinsic data structure analysis explanation. In [18], the following categories of explanations are introduced: methods that rank anomalies, methods that reveal causal relationships between anomalies, and methods that identify the attributes responsible for the abnormality of points or groups of points. In both cases, it is stated that techniques finding important attributes are the most common in the literature [6, 19]. Furthermore, while point comparison explanations focus on two points in the data set, explanations revealing cause-and-effect relationships focus on the detected anomalies, explanations by intrinsic structure analysis provide a global view on the anomaly to be explained with respect to the data set, and is thus more detailed as the whole data set context is taken into consideration. Although some works have tried to fill the gap [20, 21, 22], explanation by structure analysis lacks references.

2.3. Outlier-Aware Clustering

150 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
155 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 if the outliers had been removed from the data set, without requiring to perform a preliminary step of outlier
160 detection and removal. Robust clustering can be roughly categorized into two types of methods [23]: some of them proceed by automatically down-weighting atypical data points [24], using several approaches to define these weights, e.g. including noise clustering [25], possibilistic clustering [26, 27], replacing the traditional normal distributions by multivariate t-distributions [28] or dedicated
165 approaches [29, 30]. Other methods propose to replace the classical squared Euclidean distance, which is known to be highly sensitive to outliers, by other distances [31, 32, 33, 34, 35]. Along the same lines, some approaches are explicitly based on robust M-estimators incorporated in the cost function [36], the possibilistic c-means [26] can be seen in this framework. These approaches
170 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, RIFIFI is related to the approaches introduced in [37], [38] and [39]. The first one relies on
175 the combination of two types of clustering algorithms, a partitioning one and a hierarchical one. The last one applies a partitioning clustering algorithm to an auxiliary binary representation of the data. In [38], a robust k -means extension, called k -means- and aiming at simultaneously identifying clusters and anomalies is proposed.

180 3. The RIFIFI Approach

This section details the contribution of this article as an extension of the initial IF algorithm (Alg. 1) to isolate anomalies but also to reconstruct the regular points inner structure. The principle of RIFIFI is first introduced before
185 detailing the proposed algorithm and how the knowledge it generates is used to isolate outliers and identify dense subspaces as well.

3.1. Principle

RIFIFI differs from classical IF (as recalled in Sect. 2) on the recursive splits/separations generation, where a split is defined as the couple (A, v) containing the chosen attribute and value: while classical IF uses completely random separations, RIFIFI has a split selection criterion based on the density of
190

Table 1: Notations used throughout the paper

| Notation | Meaning |
|-------------------------------------|---|
| \mathcal{D} | Data set of n points |
| $\mathcal{A} = \{A_1, \dots, A_m\}$ | Descriptive attributes |
| $dom(A)$ | Domain of attribute A |
| I^a | Interval on feature $a \in \mathcal{A}$ |
| $x \in \mathcal{D}$ | Data point |
| $x.a$ | Value for data point x on attribute a |
| t | Nb. of trees in the forest denoted by $\mathcal{F} = \{T_1, \dots, T_t\}$ |
| Ψ | Cardinality of the data subset used to build a tree |
| h_{lim} | Depth limit |
| $n_i(x)$ | Cardinality of the node containing x in the i -th tree |
| α | Margin width |
| η | Density threshold |

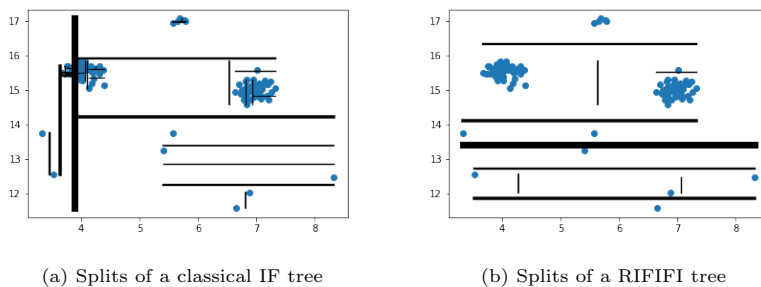


Figure 2: Examples of splits, shown by the black lines, of a tree: (left) IF, (right) RIFIFI. The width of the line is inversely proportional to the depth of the split.

the subspace in the neighborhood of each split. 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. Another split must therefore be generated. The goal is to surround the regular point clusters by the separations, so that

195 leaves may contain a cluster, or a significant portion of a cluster. Two hyper-parameters are introduced in addition to the IF hyper-parameters: the size of the margin α around the separation which represents its neighborhood, and the density threshold η . If a fraction η of points fall in the margin around the split, it is discarded.

200 The impact of this criterion on the isolation procedure is illustrated in Figure 2 that depicts two examples of trees. With the proposed criterion, the separations more rarely separate points belonging to the same cluster, and the anomalies remain isolated. However, since sampling is still performed during the construction of the trees, some separations may still separate points belonging

205 to the same cluster. In this case, several leaves may contain portions of a same cluster that have to be combined to reconstruct the whole data inner structure.

3.2. RIFIFI Algorithm

Algorithm 2 presents the details of RIFIFI. To avoid generating separations in intervals that have already been discarded because they contain many data points, the set of tested intervals I (I^a being the intervals on attribute a) is stored and passed as a parameter through the recursive calls to the *build_tree* function (lines 17 and 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 node is returned (line 4), the current set of points being considered as inseparable.

In comparison to a classical isolation forest, a RIFIFI forest induces an additional cost related to the storage of the excluded intervals. This overhead is in the worst case a constant equal to $|\mathcal{A}| * (100/\alpha + 1)$. The time complexity differs from that of a classical isolation forest by the selection of the separations. This difference is, in the worst case, linear with respect to the number of attributes: $\mathcal{O}(|\mathcal{A}|)$.

3.3. Types of Leaves Generated by RIFIFI

We propose to distinguish between three types of terminal nodes in a tree generated by Algorithm 2, depending on the condition of the stopping criterion it satisfies: 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 $C = \mathcal{A}$ (1.3 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, RIFIFI also creates nodes of type DN that are particularly informative in the prospect of reconstructing the data inner structure.

3.4. RIFIFI for Anomaly Detection

In the original IF approach, the anomaly score of a point to evaluate depends on its depth of isolation in the different trees of the forest. Relying on a completely random nested separation strategy, anomalies are those points that are the most quickly isolated and thus that appear at the top of the isolation trees in nodes of type IN. In the RIFIFI approach, and as detailed in Algorithm 2, the construction of the isolation trees still relies on randomly chosen separation lines but that may be discarded according to a density constraint. Leaves of the obtained trees may thus contain a single point isolated after a sequence of separations (leaves of type IN) or a group of points that remain unseparated at the end of the tree construction process. This may happen for two reasons: the tree depth threshold is reached (leaves of type DLN) or the whole domain of all the attributes has been explored (leaves of type DN).

Due to the fact that separations cannot split dense areas anymore, a leaf of type DN containing a high number of inseparable points can be located at

Algorithm 2 RIFIFI : *build_tree*

```

1: Inputs: data sample  $D \subset \mathcal{D}$ , depth  $d$  of the current node, margin width
    $\alpha$ , density threshold  $\eta$ , sets of tested intervals  $I = \{I^{A_1}, \dots, I^{A_m}\}$ , set of
   covered attributes  $C$ 
2: Output: a node in an isolation tree
3: if  $C = \mathcal{A}$  or  $|D| = 1$  or  $d > h_{lim}$  then
4:   Return  $node(null, null, D, d, null, null)$  ▷ Leaf
5: else
6:    $a \leftarrow random(\mathcal{A} \setminus C)$  ▷ Random selection of an attribute among
   ▷ the untested attributes
7:    $v \leftarrow random(domain(a) \setminus \cup_J \{J \in I^a\})$  ▷ Random selection of a value
   ▷ among the untested values for that attribute
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  $[\min_{x \in D} x.a, \max_{x \in D} x.a] \subseteq I^a$  then ▷ The entire range of values
   ▷ has been scanned and excluded
11:     $C \leftarrow C \cup \{a\}$  ▷ The attribute is added to the list
   ▷ of excluded attributes
12:  end if
13:   $D_m \leftarrow \{x \in D/x.a \in [v - marg, v + marg]\}$  ▷ Points contained in the
   ▷ margin
14:  if  $|D_m| \leq \eta$  then
15:     $D_l \leftarrow \{x \in D/x.a < v\}$ 
16:     $D_r \leftarrow \{x \in D/x.a \geq v\}$ 
17:    Return  $node(build\_tree(D_l, d + 1, \alpha, \eta, \emptyset, \emptyset),$  ▷ Internal node
18:              $build\_tree(D_r, d + 1, \alpha, \eta, \emptyset, \emptyset), D, d, a, v)$ 
19:  end if
20:  Return  $build\_tree(D, d, \alpha, \eta, I, C)$  ▷ Selection of another split
21: end if

```

250 low depth. It thus makes more sense to define an anomaly score function that depends on the number of points grouped together in a leaf, instead of its depth. Hence the proposed anomaly scoring function:

$$s_i(x) = 1 - \frac{n_i(x) - 1}{\Psi}. \quad (2)$$

where $n_i(x)$ denotes the cardinality of the node containing x in the i -th tree. The score $s_i(x)$ varies in $]0, 1]$, it takes its maximum value ($s_i(x) = 1$) 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 last situation occurs when, according to the density threshold η and the margin α , no separation line can be validated on the whole universe: the dataset consists of a single indivisible cluster.

255

The global anomaly score at the forest level is the average on all t trees:

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

260 With original IF, the anomaly score of a data point reflects the ease (in terms of splits) with which it is separated from others. With LOF [8], the abnormality of a data point is caused by its lower surrounding density compared to its neighbors. As RIFIFI splits are located in low density regions, the method combines the separability property used in IF and the local density property
 265 used in LOF to identify anomalies.

3.5. RIFIFI for Clustering

Whereas the anomaly detection step leverages leaves of type IN and possibly of type DN when they gather very small subsets of inseparable anomalies, the reconstruction of the data inner structure uses leaves of type DN only. As
 270 defined in Section 3.3, leaves of type DN correspond to dense regions of points that may form an elliptic cluster or a part of any shape of cluster. Using Algorithm 2, one knows that points grouped in a DN cannot be separated anymore, according to the considered hyper-parameter values for α (margin width around the separation lines) and η (density threshold).

275 Therefore, still in the spirit of ensemble-based approaches, if points are frequently found in the same DN in the different trees, they probably belong to the same cluster. We thus define an **inseparability index** between two points as the average number of times they co-occur in the same DN:

$$sim(x_1, x_2) = \frac{1}{t} \sum_{f \in \mathcal{F}} \mathbb{1}_f(x_1, x_2) \quad (4)$$

with \mathcal{F} the set of leaves of type DN in the forest, and $\mathbb{1}_f(x_1, x_2) = 1$ if $\{x_1, x_2\} \subseteq f$ and 0 otherwise.
 280

The points can then be combined progressively on the basis of their inseparability index using Agglomerative Hierarchical Clustering (AHC) to reconstruct a partition of the data set.

4. Experiments

285 The goal of this section is two-fold: show that RIFIFI is still able to correctly identify the anomalies, and show that RIFIFI preserves the structure of regular data. The latter is equivalent to checking whether each leaf is a portion of a cluster and the information contained in the leaves can be used to reconstruct the structure of regular data.

290 Throughout the experiments, default values are set for parameters of RIFIFI: $t = 100$, $\Psi = 256$, $\alpha = 0.05$ and $\eta = 0.5$. The values of t and Ψ are identical to the default values of the classical IF [4]. A fixed margin width of $\alpha = 5\%$

Table 2: Considered anomaly detection data sets (available in [40]): dimension, number of instances and number of anomalies.

| Name | d | n | # of anomalies |
|-------------|-----|--------|----------------|
| Anthyroid | 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 |

of the attribute initial range is chosen. The intuition behind a fixed value of α is the following : if two points are separated by less than that $\alpha * range(a)$ on an attribute a , they should remain together during the tree building process. However, 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 $\beta/range(a)$. The motivation for the choice of η is the following: if the data points are uniformly distributed, then $n' = \alpha \times N'$ points should fall within the margin, where N' is the number of points in the current node. As a result, if less than $0.5 \times n'$ data points fall within the margin, one can assume that the separation is not splitting a group of close points.

4.1. Anomaly Detection

The objective of this part of the experiments is to evaluate the anomaly detection component of RIFIFI. In the presented experimentations, RIFIFI is compared with existing approaches according to its capability of separating outliers from regular points.

Data sets

Thirteen data sets, including 2 statistical ones, are used, that are similar to the ones considered in [4] to evaluate IF. The dimension, number of instances and number of anomalies of each data set are presented in Table 2. The expected anomalies are known for each data set and serve as ground truths during the evaluation.

General Assessment: Area Under Curves

For each data set, the Area Under the Receiver Operating Characteristic curve (AUC or AUROC) of RIFIFI and IF are compared. This metric is gen-

Table 3: Mean AUC and standard deviations of IF, RIFIFI, EIF, SCIForest and FCF.

| Data set | IF | RIFIFI | EIF | SCIForest | FCF |
|-------------|------------------|----------------------|--------------------|--------------------|--------------------|
| Anthyroid | 0.802 ± 0.016 | 0.778 ± 0.022 | 0.716 ± 0.0 | 0.760 ± 0.0 | 0.881 ± 0.0 |
| Arrhythmia | 0.765 ± 0.027 | 0.822 ± 0.007 | 0.813 ± 0.0 | 0.683 ± 0.0 | 0.809 ± 0.0 |
| Breast | 0.979 ± 0.003 | 0.992 ± 0.001 | 0.987 ± 0.0 | 0.983 ± 0.0 | 0.984 ± 0.0 |
| Cover | 0.885 ± 0.02 | 0.857 ± 0.02 | 0.904 ± 0.0 | 0.704 ± 0.0 | 0.929 ± 0.0 |
| Hbk | 1.0 ± 0.0 | 1.0 ± 0.0 | 1.0 ± 0.0 | 1.0 ± 0.0 | 0.984 ± 0.0 |
| Http | 0.999 ± 0.001 | 0.997 ± 0.002 | 0.999 ± 0.0 | 0.999 ± 0.0 | 0.999 ± 0.0 |
| Ionosphere | 0.847 ± 0.007 | 0.832 ± 0.005 | 0.843 ± 0.0 | 0.890 ± 0.0 | 0.872 ± 0.0 |
| Mammography | 0.610 ± 0.03 | 0.843 ± 0.005 | 0.869 ± 0.0 | 0.585 ± 0.0 | 0.655 ± 0.0 |
| Pima | 0.676 ± 0.01 | 0.683 ± 0.007 | 0.676 ± 0.0 | 0.600 ± 0.0 | 0.662 ± 0.0 |
| Satellite | 0.705 ± 0.012 | 0.685 ± 0.009 | 0.694 ± 0.0 | 0.623 ± 0.0 | 0.718 ± 0.0 |
| Shuttle | 0.994 ± 0.001 | 0.993 ± 0.001 | 0.993 ± 0.0 | 0.997 ± 0.0 | 0.994 ± 0.0 |
| Smtp | 0.891 ± 0.007 | 0.866 ± 0.008 | 0.868 ± 0.0 | 0.935 ± 0.0 | 0.925 ± 0.0 |
| Wood | 0.903 ± 0.04 | 1.0 ± 0.0 | 0.843 ± 0.0 | 1.0 ± 0.0 | 0.953 ± 0.0 |
| Mean AUC | 0.85 | 0.88 | 0.86 | 0.82 | 0.87 |

erally used for the evaluation of anomaly detection methods because it is independent of the anomaly score threshold. It represents the probability that anomalies receive higher scores than regular instances. The AUCs of Extended Isolation Forest (EIF) [14], SCIForest [13] and Fair Cut Forest (FCF) [15] on the data sets are also computed for comparison. The implementations of EIF, SCIForest and FCF are available in the package *isotree*².

Table 3 reports the mean AUC obtained on each data set after 10 runs of each method, and the associated standard deviations.

IF performs better than RIFIFI on 7 data sets, and RIFIFI performs better than IF on 5 data sets. Both methods obtain the same perfect results on the statistical data set *hbk*. In most data sets, there is no significant difference between RIFIFI and IF. However, on the data set *mammography*, RIFIFI performs much better than IF, with a gain of +0.233 in mean AUC. In average, RIFIFI performs better than the classical IF, with an average gain of +0.03 in AUC. Considering the big picture, the different variants display similar performances.

Identified Anomalies

Why does RIFIFI performs better than IF on some data sets? The answer to this question has two parts. The first part can be observed on the statistical data set *wood*. On this data set and for this batch of experiments, RIFIFI systematically assigns a higher score to the real anomalies in comparison to IF. It is not the case for IF. Table 4 shows the 10 data points that receive the highest anomaly scores for both methods. Using the same representation as in [4], Figure 3 shows the first two principal components of the data set.

The four highest-ranked instances by RIFIFI are the actual anomalies of the data set (instances 4, 6, 8 and 19), whereas IF scores the instance 10 first. Observing the two principal components on Figure 3 shows that instance 10, although regular, lies in a low density subspace. Since with IF the anomaly score only depends on the average isolation depth of a data point (Eq. 1), it is more difficult for the method to make a distinction between real anomalies and regular data points located in low density subspaces. On the other hand, RIFIFI takes the local density of the data point into consideration while generating the

²<https://github.com/david-cortes/isotree/blob/master/README.md>

Table 4: Anomaly ranking of the *wood* data set: bold-faced indexes are actual anomalies.

| Rank | IF | RIFIFI |
|------|-----------|-----------|
| 1 | 10 | 19 |
| 2 | 19 | 4 |
| 3 | 4 | 6 |
| 4 | 8 | 8 |
| 5 | 20 | 7 |
| 6 | 7 | 12 |
| 7 | 1 | 11 |
| 8 | 12 | 9 |
| 9 | 11 | 1 |
| 10 | 17 | 20 |

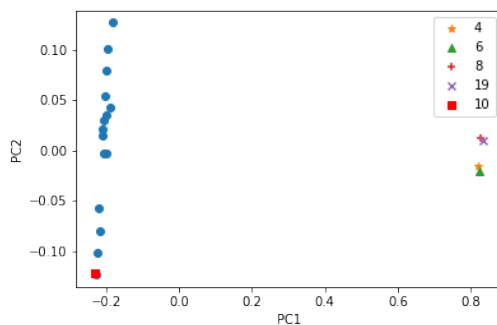


Figure 3: First two principal components of the wood data set. Instances 4, 6, 8, 10 and 19 are displayed.

splits and during the score computation (Eq. 2). As a result, RIFIFI offers a
 350 better contrast between regular data points located in low density subspaces
 (but still surrounded by data points when the local density is considered) and
 anomalies (isolated). LOF also correctly identifies the true anomalies, and rank
 them first [4]. With LOF, there is also no ambiguity because instance 10 has a
 surrounding density similar to the one of its neighbors. These findings are in line
 355 with the principle stated at the end of section 3.4: RIFIFI combines separability
-anomalies are far from the other data points- and local density information -
anomalies have a lower local density in comparison to their neighbors- during
 the identification of anomalies.

The second difference between IF and RIFIFI lies in the identification of
 360 local anomalies. This phenomenon can be observed on the data set on Figure 4.
 On this figure, the opacity of each data point is proportional to its anomaly
 score. The scores are min-max scaled. It appears that RIFIFI gives higher
 scores than IF to local anomalies.

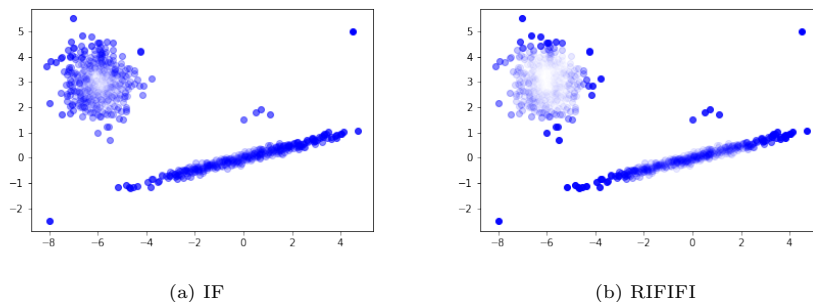


Figure 4: Scores distribution: IF vs RIFIFI

Table 5: Minimum, maximum and mean AUC values obtained when varying the hyper-parameters

| Data set | Min | Mean | Max |
|-------------|-------|-------|-------|
| Anthyroid | 0.752 | 0.769 | 0.782 |
| Arrhythmia | 0.810 | 0.814 | 0.819 |
| Breast | 0.993 | 0.994 | 0.995 |
| Cover | 0.819 | 0.852 | 0.889 |
| Hbk | 1.0 | 1.0 | 1.0 |
| Http | 0.993 | 0.996 | 0.998 |
| Ionosphere | 0.826 | 0.830 | 0.836 |
| Mammography | 0.826 | 0.841 | 0.848 |
| Pima | 0.682 | 0.694 | 0.717 |
| Satellite | 0.680 | 0.691 | 0.700 |
| Shuttle | 0.992 | 0.993 | 0.993 |
| Smtpt | 0.871 | 0.886 | 0.900 |
| Wood | 0.941 | 0.960 | 0.981 |

Hyper-parameters Sensitivity

365 How much does the choice of the hyper-parameters influence the anomaly detection performance of RIFIFI?

As a reminder, RIFIFI introduces two additional hyper-parameters: α which represents the width of the margin surrounding the split and η which controls the density surrounding of the margin during the split selection. They have the following default values: $\alpha = 5\%$ of the attribute initial range, and $\eta = 0.5$.

370 For different values of $\alpha \in \{2.5\%, 5\%, 7.5\%\}$ and $\eta \in \{0.5, 1.0\}$, and for each data set, ten RIFIFI forests are built. The mean, minimum and maximum values among the $10 \times 3 \times 2$ different combinations are displayed on Figure 5 and shown in Table 5. The means and standard deviations of IF and RIFIFI using default parameters are also illustrated. They have already been shown earlier in Table 3.

It appears that the AUCs do not vary much with the parameters, and that

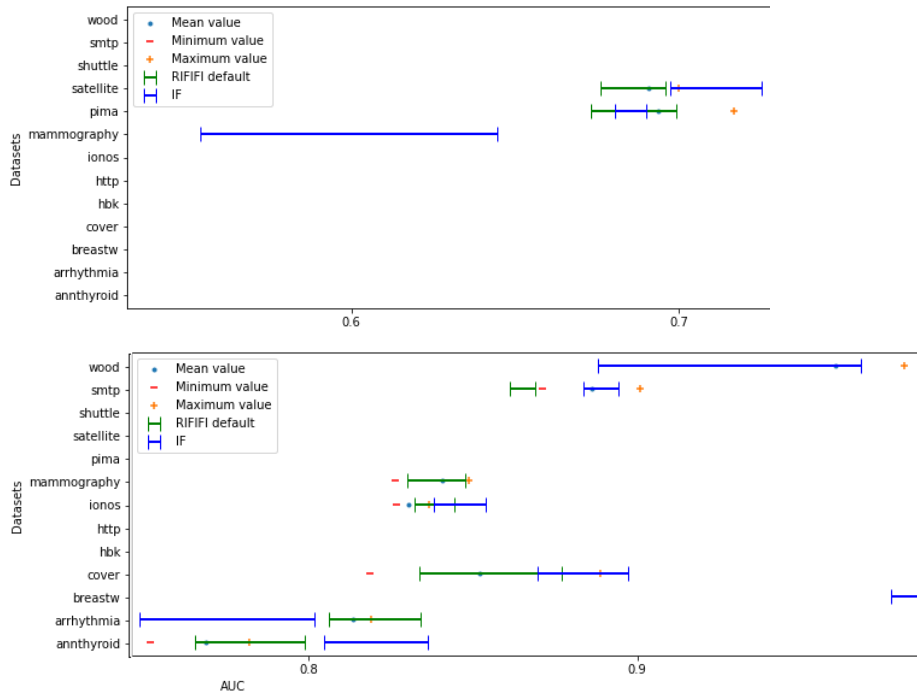


Figure 5: Influence of the hyperparameters values on the AUC

the default values are appropriate. It also appears that the anomaly detection performance of RIFIFI presents less variability than that of IF, as highlighted by the smaller standard deviations in the former. This is due to the fact that some properties of RIFIFI are controlled in a deterministic manner. Consequently, though RIFIFI remains a random method, it is less random than classical IF.

4.2. Clustering: Identifying the Inner Data Structure

The objective of this part is to investigate whether a relevant partition of the regular data points can be inferred from a RIFIFI forest. The experiments described in this section aims at checking that RIFIFI is able to reconstruct the groups of points defined in the reference corpus.

Considered Datasets

The data sets used in this section are illustrated on Figure 6. They are constrained to 2D and 3D description spaces so as to control the behavior of the method. Each of them contains clusters and anomalies: 2 clusters of regular data for \mathcal{D}_1 , \mathcal{D}_2 and \mathcal{D}_4 , 3 clusters of regular data for \mathcal{D}_3 and 4 clusters of regular data for \mathcal{D}_5 . \mathcal{D}_5 is a three-dimensional data set in which each cluster is located in a 2-dimensional subspace [41]. \mathcal{D}_4 is the data set *moons* composed of two interleaving half circles, to which anomalies have been added manually.

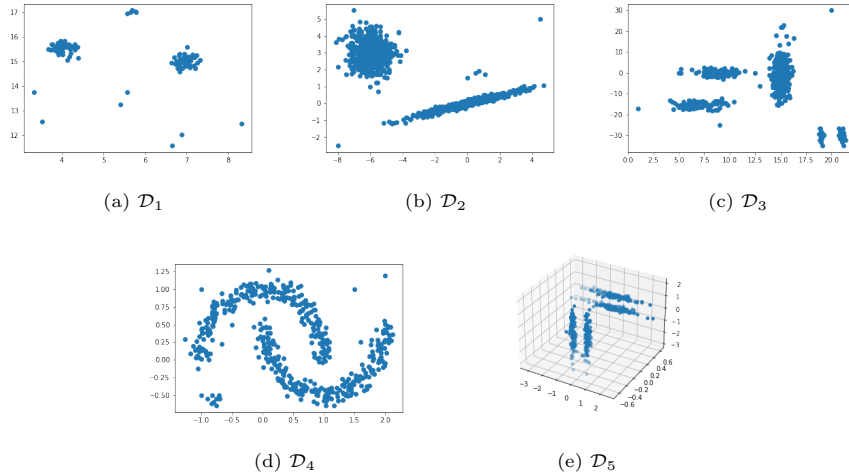


Figure 6: Considered 2D and 3D data sets for the clustering experiments

| Data set | IF | RIFIFI |
|-----------------|-------------------------------------|-------------------------------------|
| \mathcal{D}_1 | 0.999 ± 0.001 | 1.0 ± 0.0 |
| \mathcal{D}_2 | 0.980 ± 0.003 | 0.986 ± 0.002 |
| \mathcal{D}_3 | 0.966 ± 0.006 | 0.960 ± 0.008 |
| \mathcal{D}_4 | 0.998 ± 0.002 | 0.999 ± 0.0 |

Table 6: Mean AUC and standard deviations of IF and RIFIFI.

Anomaly Detection

Before diving into the clustering experiments, the anomaly detection performances of IF and RIFIFI on these data sets are compared to confirm that RIFIFI addresses the two tasks, namely anomaly detection and clustering, using a unified data model, i.e. the isolation forest. Table 6 reports the mean AUCs after 10 runs and the associated standard deviations. The results on \mathcal{D}_5 are not shown because anomalies were not manually added to this data set during its generation. All the instances are expected to be regular, even though some instances are slightly deviating as part of the generation process.

Leaf Cardinalities and Tree Depths

The impact of the split selection in RIFIFI on the size of the leaves is evaluated. With IF, the separations are completely random until a point is isolated or the depth limit is reached. It is therefore expected to have on one hand leaves containing isolated points, and on the other hand deeper leaves containing more than a single point. With RIFIFI, it is expected to have leaves containing isolated points, leaves containing points that could not be separated and leaves that have reached the depth limit. Ideally, there should be more leaves of the

| Data set | Leaf sizes | | Tree depths | |
|-----------------|------------------|------------------|-----------------|-----------------|
| | IF | RIFIFI | IF | RIFIFI |
| \mathcal{D}_1 | 7.30 ± 0.45 | 30.29 ± 0.84 | 8.0 ± 0.0 | 4.36 ± 1.51 |
| \mathcal{D}_2 | 8.76 ± 0.24 | 45.02 ± 1.37 | 8.0 ± 0.0 | 4.96 ± 1.03 |
| \mathcal{D}_3 | 10.10 ± 0.36 | 25.68 ± 0.67 | 8.0 ± 0.0 | 5.98 ± 0.96 |
| \mathcal{D}_4 | 6.65 ± 0.14 | 38.10 ± 1.35 | 7.98 ± 0.02 | 4.21 ± 1.30 |
| \mathcal{D}_5 | 9.22 ± 0.46 | 17.49 ± 0.84 | 8.0 ± 0.0 | 7.11 ± 0.83 |

Table 7: Statistics on the tree structures built by IF and RIFIFI

| Data set | DLN leaves (%) | DN leaves (%) |
|-----------------|------------------|------------------|
| \mathcal{D}_1 | 22.63 ± 6.96 | 77.37 ± 6.96 |
| \mathcal{D}_2 | 30.85 ± 5.64 | 69.15 ± 5.64 |
| \mathcal{D}_3 | 35.25 ± 3.99 | 64.75 ± 3.99 |
| \mathcal{D}_4 | 12.19 ± 3.21 | 87.81 ± 3.21 |
| \mathcal{D}_5 | 70.02 ± 3.26 | 29.98 ± 3.26 |

Table 8: Percentages of the different types of leaves

second type, depending on the chosen depth limit, because the objective is to preserve the clusters. Therefore, RIFIFI trees should be shallower (the depth limit being more difficult to reach than in the classical version) and the leaves should contain more data points.

A classical Isolation Forest and a RIFIFI forest are built on each data set. The leaves containing isolated instances (IN leaves) are discarded. Then, the average cardinality of the leaves as well as the average depths of the trees of each forest type are computed. The means and standard deviations across 10 runs are reported in Table 7.

RIFIFI leaves contain more points than the leaves of a classical isolation forest, and this on all the data sets. As for the trees, they are shallower than the classical isolation trees.

Types of Leaves

We then study the proportion of leaves that have reached the depth limit (DLN), as compared to the proportion of leaves containing points that are no longer separable (DN): a RIFIFI forest is built and these two values are computed. Table 8 reports the means and standard deviations across 10 runs of this experiment.

It appears that a significant proportion of leaves are DN. This phenomenon is verified on data sets \mathcal{D}_1 to \mathcal{D}_4 , but not on data set \mathcal{D}_5 . The latter also contains fewer points in the leaves, as compared to the other data sets and the trees of the RIFIFI forest, although shallower than the classical isolation trees, are still deeper than those of the forests built on the other data sets (Table 7). This is explained by the fact that in \mathcal{D}_5 , each cluster "exists" in only two of

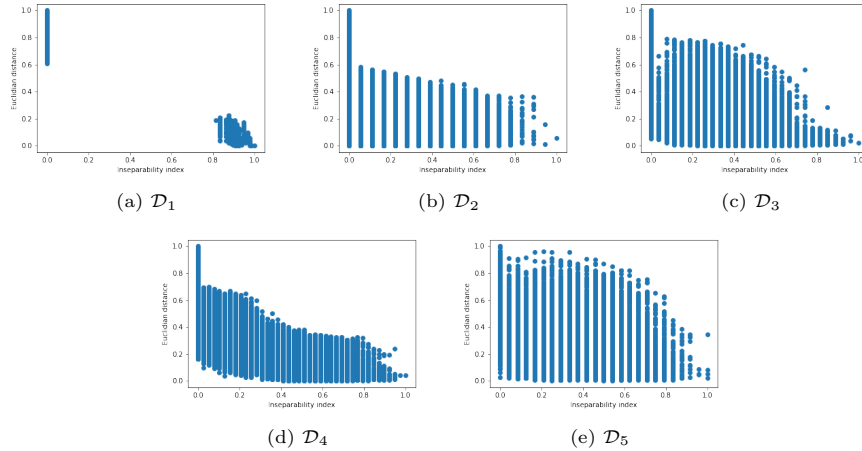


Figure 7: Euclidean distance vs inseparability index

the three dimensions. However, the isolation process continues by separating the points on the third dimension, where they are distributed almost uniformly. The depth limit is not a function of the number of dimensions. However, as the dimension of the data set increases, there are more options for the split choice. As a result, the depth limit is more often reached. Increasing the depth limit taking into account the dimensionality mitigates the aforementioned problem. For example, by using a depth limit of 15 for \mathcal{D}_5 , the percentage of DLN leaves decreases to 33.27 ± 2.47 .

445 *Data Point Proximity*

This part intends to check if RIFIFI preserves the proximity between the data points. This proximity is measured in the original data space by the Euclidean distance and in RIFIFI by the inseparability index (Eq. 4). For each pair of points in the data set, the Euclidean distance between them is calculated, as well as the inseparability index. Both values are min-max scaled. The results are displayed for each data set on Figure 7: for each pair of points, on the x -axis the inseparability index and on the y -axis the Euclidean distance.

Two seemingly counter-intuitive phenomena are observed when analyzing these results, but can be explained as follows. First, some data points, despite being close in Euclidean space (small Euclidean distance), are rarely found in the same leaf (inseparability index close to 0). This occurs when the two points, although close in the Euclidean space, are separable and thus belong to different clusters, for example the instances $(14.22, -0.70)$ and $(8.59, -0.89)$ in \mathcal{D}_3 . It is especially the case when they have similar values on some dimensions. A separation between these two points could be kept if their neighborhood is not dense. The aforementioned situation also occurs when one of the two points is very close to the cluster containing the other point, without being part of it, or when one of the two points is located at the border of the cluster and is therefore

| | Average distance within leaves | | Average distance between leaves | |
|-----------------|---------------------------------|---------------------------------|---------------------------------|----------------------------------|
| | IF | RIFIFI | IF | RIFIFI |
| \mathcal{D}_1 | 0.102 \pm 0.002 | 0.163 \pm 0.002 | 1.684 \pm 0.007 | 2.523 \pm 0.033 |
| \mathcal{D}_2 | 0.233 \pm 0.004 | 0.507 \pm 0.012 | 4.114 \pm 0.034 | 5.230 \pm 0.040 |
| \mathcal{D}_3 | 1.142 \pm 0.037 | 1.465 \pm 0.055 | 13.250 \pm 0.182 | 19.070 \pm 0.253 |
| \mathcal{D}_4 | 0.117 \pm 0.004 | 0.237 \pm 0.005 | 1.256 \pm 0.006 | 1.474 \pm 0.012 |
| \mathcal{D}_5 | 0.341 \pm 0.008 | 0.299 \pm 0.014 | 1.419 \pm 0.009 | 1.598 \pm 0.027 |

Table 9: Average distances within and between leaves, means and standard deviations across 10 runs

often separated from the others (e.g. points $(-7.10; 4.57)$ and $(-6.03; 3.16)$ in \mathcal{D}_2).

It can also be observed that some data points distant in the Euclidean space are sometimes found in the same leaves. This occurs when the two points, although distant in Euclidean space, are part of the same cluster, for example when the cluster is stretched. This phenomenon frequently occurs in the data set \mathcal{D}_5 where all the four clusters are stretched.

This analysis suggests that three random points x_1 and x_2 then x_1 and x_3 can be located at the same Euclidean distance, but, using the information provided by the RIFIFI forest, x_1 and x_2 are part of the same cluster, and x_3 is not, because many splits separate x_1 and x_3 . The local density evaluation during the split selection therefore brings an additional knowledge useful for clustering.

Average Distances within and between Leaves

This section of the experiments aims at checking whether RIFIFI’s leaves are portions of clusters containing close points which are separable from other leaves not part of the same cluster. To verify that, the average Euclidean distance between the points of each leaf and the center of the leaf is calculated for both forest types. It is the average distance within leaves. The average Euclidean distance between the centers of the leaves is also computed. The results are reported in Table 9.

The average distance within leaves is larger in RIFIFI in almost all datasets, which is understandable because classical isolation leaves contain less points as seen earlier in the experiments, and these data points are close. RIFIFI’s leaves in contrast contain larger groups of close data points. On \mathcal{D}_5 , the average distance within leaves is larger in IF: for data points belonging to two clusters located in different subspaces, the distance is much larger. On the other hand, the distance between leaves is systematically larger in RIFIFI, which reflects the fact that there is a better separability between leaves on RIFIFI, in comparison to IF.

| Data set | Inseparability index | Euclidean distance |
|-----------------|----------------------|--------------------|
| \mathcal{D}_1 | 1.0 | 1.0 |
| \mathcal{D}_2 | 0.953 | 0.957 |
| \mathcal{D}_3 | 0.958 | 0.429 |
| \mathcal{D}_4 | 0.645 | 0.387 |
| \mathcal{D}_5 | 0.855 | 0.588 |

Table 10: AHC on the data sets using inseparability index vs Euclidean distance: Adjusted Rand Indexes

Inseparability Index and Clustering

495 The purpose of this experiment is to verify whether, when two points have a low inseparability index, they indeed belong to the same cluster.

For each data set, a RIFIFI forest is built and anomalies are identified. The anomaly score threshold is set to 0.90³. Then, the inseparability index between each pair of points is computed and an Agglomerative Hierarchical Clustering using average linkage is performed on the obtained similarity matrix. As the number k of clusters in the data set is known, the AHC is stopped when k groups are constructed.

500 The obtained clusters are compared to the expected ones using the *Adjusted Rand Index* (ARI), that equals 1 if the two partitions are identical. Table 10 shows the maximum ARI obtained on each data set, compared with the maximum ARI obtained when using the Euclidean distance as the distance measure for the AHC. In the ARI calculation, anomalies are considered as part of an isolated cluster.

510 Except on data sets \mathcal{D}_1 and \mathcal{D}_2 , the ARI is higher when using the inseparability index. On \mathcal{D}_3 , the separability information conveyed by the inseparability index allows to reconstruct the three regular clusters, where the Euclidean distance combines the upper-half of the biggest cluster to the cluster at the top left (Fig. 8a).

515 As DN leaves are either elliptic clusters or parts of clusters, their combination (in the case of cluster parts) can lead to the discovery of non elliptic clusters. This is observed on data set \mathcal{D}_4 . The performance on this data set is nevertheless limited because of the value of η : some splits group together a part of the lower half-moon and the portion of the upper half-moon located in the cavity of the first mentioned (Fig. 9b). As a result, these upper half-moon points are assigned to the same cluster as the lower half-moon points (Fig. 9c). When the density condition is relaxed, with $\eta = 1$, the ARI reaches 0.95 (Fig. 9d).

520 On \mathcal{D}_5 , the clusters are stretched and located in different subspaces. Consequently, and as observed on Figure 7e, points belonging to the same cluster may

³In practical anomaly detection, the user can either choose a threshold or consider as abnormal the p instances which receive the highest anomaly score, where p is a *small* percentage of the data set.

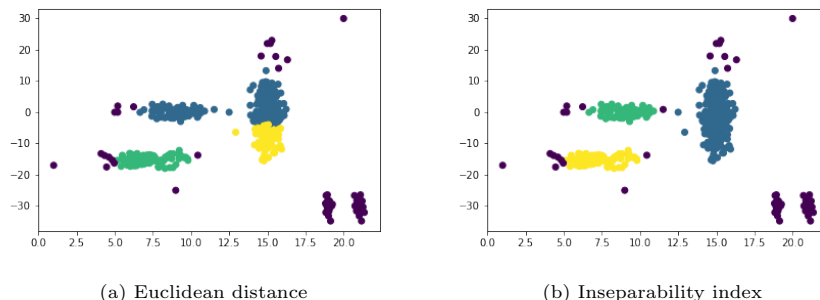


Figure 8: AHC on \mathcal{D}_3 : Euclidean distance vs inseparability index

be far away from each other when considering the Euclidean distance. The
 525 inseparability index however is not tricked by that subtlety of the data set because
 the instances are frequently located in the same DN leaves. On the other hand,
 points belonging to different clusters are sometimes close because the clusters
 are located in the same subspace. Again, where the combination Euclidean distance + AHC merges those two clusters, the inseparability index is able to keep
 530 them separated. Figure 10 illustrates these results.

5. Conclusion and perspectives

In a data to knowledge translation process, providing users with informative
 explanations about the dataset inner structure and the presence of anomalies is
 an ultimate goal. This work makes a step towards this objective targeting the
 535 specific task of differentiating global and local anomalies. Whereas explaining
 the reason why a point constitutes a global anomaly is not that difficult, it is
 generally due to the existence of unseen extreme values the point possessed on
 a subset of attributes, local anomalies requires a better understanding of the
 data set. An anomaly is said to be local to a group of regular points if it shares
 540 some characteristic values of these regular points but also possesses values not
 observed in this group. This distinction between global and local anomalies
 cannot be made without knowing the data inner structure.

This paper proposes a variant of the isolation forest algorithm called RIFIFI
 with the objective of preserving the clusters located in the data set. For this
 545 purpose, a new criterion for the selection of separations has been introduced,
 based on the analysis of the neighborhood of the separations. The first carried
 out experiments show that the proximity between points belonging to the same
 group of data can be preserved, and that the reconstitution of a partition of
 the data set is thus possible by carrying out an Agglomerative Hierarchical
 550 Clustering step on a similarity matrix based on the number of times that the
 pairs of points are found in the same leaf.

This work is a first step towards a unified approach for extracting contextual
 explanations of anomalies. Indeed, the ideal setting would be to use directly the

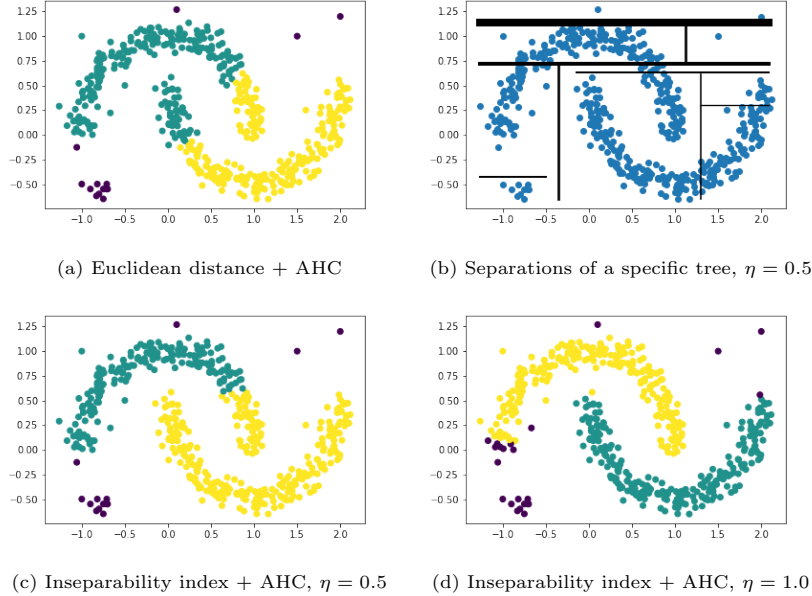


Figure 9: AHC on \mathcal{D}_4 and impact of η : (top left) reference result obtained with the Euclidean distance, (top right) example of a tree for $\eta = 0.5$, (bottom left) partition obtained with the inseparability index with $\eta = 0.5$, (bottom right) partition obtained with the inseparability index with $\eta = 1$.

information contained in the leaves, without computing the distances (Euclidean
555 or not) between pairs of points, since isolation forests do not require these
calculations. To do so, an aggregation of the leaves similar to clustering methods
of the type *grid-based* could be explored: each leaf of significant cardinality
delimits a subspace, and the different subspaces can be combined to reconstitute
a partition of the data set. Having the anomalies on one side, and this partition
560 on the other, it would become possible to extract contrastive explanations of
anomalies using a unified method, without relying on a pipeline.

Acknowledgement

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

References

- [1] I. Santos, C. Laorden, X. Ugarte-Pedrero, B. Sanz, P. G. Bringas, Spam
filtering through anomaly detection, in: E-Business and Telecommunica-
tions, Springer Berlin Heidelberg, 2012, pp. 203–216.

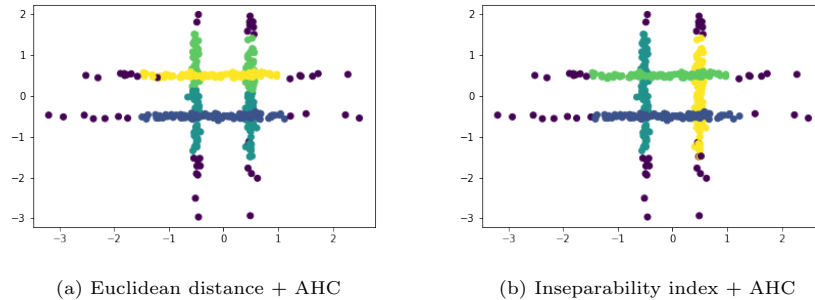


Figure 10: AHC results on \mathcal{D}_5

- 570 [2] R. Hou, Y. Peng, L. J. Grimm, Y. Ren, M. A. Mazurowski, J. R. Marks, L. M. King, C. C. Maley, E. S. Hwang, J. Y. Lo, Anomaly detection of calcifications in mammography based on 11,000 negative cases, *IEEE Transactions on Biomedical Engineering* 69 (2022) 1639–1650.
- [3] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, *ACM Comput. Surv.* 41 (2009).
- 575 [4] F. T. Liu, K. M. Ting, Z.-H. Zhou, Isolation-based anomaly detection, *ACM Trans. Knowl. Discov. Data* 6 (2012).
- [5] M. Kopp, T. Pevný, M. Holeňa, Anomaly explanation with random forests, *Expert Systems with Applications* 149 (2020).
- [6] T. Mokoena, T. Celik, V. Marivate, Why is this an anomaly? explaining anomalies using sequential explanations, *Pattern Recognition* 121 (2022).
- 580 [7] V. Yepmo, G. Smits, O. Pivert, Anomaly explanation: A review, *Data & Knowledge Engineering* 137 (2022).
- [8] M. M. Breunig, H.-P. Kriegel, R. T. Ng, J. Sander, Lof: Identifying density-based local outliers, in: *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, 2000, pp. 93–104.
- 585 [9] M. Amer, M. Goldstein, S. Abdennadher, Enhancing one-class support vector machines for unsupervised anomaly detection, in: *Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description*, 2013, pp. 8–15.
- 590 [10] M. Sakurada, T. Yairi, Anomaly detection using autoencoders with non-linear dimensionality reduction, in: *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, 2014, pp. 4–11.
- [11] F. Angiulli, F. Fassetti, C. Serrao, Anomaly detection with correlation laws, *Data & Knowledge Engineering* 145 (2023).

- 595 [12] A. Mensi, M. Bicego, Enhanced anomaly scores for isolation forests, *Pattern Recognition* 120 (2021) 108–115.
- [13] F. T. Liu, K. M. Ting, Z.-H. Zhou, On detecting clustered anomalies using sciforest, in: *Machine Learning and Knowledge Discovery in Databases*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 274–290.
- 600 [14] S. Hariri, M. C. Kind, R. J. Brunner, Extended isolation forest, *IEEE Transactions on Knowledge and Data Engineering* 33 (2019) 1479–1489.
- [15] D. Cortes, Revisiting randomized choices in isolation forests, arXiv preprint arXiv:2110.13402 (2021).
- [16] P. Karczmarek, A. Kiersztyn, W. Pedrycz, E. Al, K-means-based isolation forest, *Knowledge-Based Systems* 195 (2020).
- 605 [17] T. G. Dietterich, An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting and randomization, *Machine learning* 32 (1998) 1–22.
- [18] E. Panjei, L. Gruenwald, E. Leal, C. Nguyen, S. Silvia, A survey on outlier explanations, *The VLDB Journal* 31 (2022) 977–1008.
- 610 [19] N. Gupta, D. Eswaran, N. Shah, L. Akoglu, C. Faloutsos, Beyond outlier detection: Lookout for pictorial explanation, in: *Machine Learning and Knowledge Discovery in Databases*, Springer International Publishing, Cham, 2019, pp. 122–138.
- [20] M.-J. Lesot, A. Revault d’Allonnes, Credit-card fraud profiling using a hybrid incremental clustering methodology, in: *Scalable Uncertainty Management*, Springer Berlin Heidelberg, 2012, pp. 325–336.
- 615 [21] M. Macha, L. Akoglu, Explaining anomalies in groups with characterizing subspace rules, *Data Mining and Knowledge Discovery* 32 (2018) 1444–1480.
- 620 [22] A. K. Shukla, G. Smits, O. Pivert, M.-J. Lesot, Explaining data regularities and anomalies, in: *2020 IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE)*, IEEE, 2020, pp. 1–8.
- [23] C. Borgelt, C. Braune, M.-J. Lesot, R. Kruse, Handling noise and outliers in fuzzy clustering, in: *Fifty Years of Fuzzy Logic and its Applications*, Springer International Publishing, Cham, 2015, pp. 315–335.
- 625 [24] F. Dotto, A. Farcomeni, L. A. García-Escudero, A. Mayo-Isar, A reweighting approach to robust clustering, *Statistics and Computing* 28 (2018) 477–493.
- 630 [25] R. N. Dave, Characterization and detection of noise in clustering, *Pattern Recognition Letters* 12 (1991) 657–664.

- [26] R. Krishnapuram, J. Keller, The possibilistic c-means algorithm: insights and recommendations, *IEEE Transactions on Fuzzy Systems* 4 (1996) 385–393.
- 635 [27] N. Pal, K. Pal, J. Keller, J. Bezdek, A possibilistic fuzzy c-means clustering algorithm, *IEEE Transactions on Fuzzy Systems* 13 (2005) 517–530.
- [28] G. J. McLachlan, D. Peel, Robust cluster analysis via mixtures of multivariate t-distributions, in: *Advances in Pattern Recognition*, Springer Berlin Heidelberg, Berlin, Heidelberg, 1998, pp. 658–666.
- 640 [29] A. Keller, Fuzzy clustering with outliers, in: *Proc. of the 19th Conf. North American Fuzzy Information Processing Society, NAFIPS’00*, IEEE Press, 2000, pp. 143–147.
- [30] F. Klawonn, F. Höppner, 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, 2003, pp. 254–264.
- 645 [31] A. Seheult, P. Green, P. Rousseeuw, A. Leroy, Robust regression and outlier detection., *Journal of the Royal Statistical Society. Series A (Statistics in Society)* 152 (1989) 133.
- [32] K. Jajuga, L1-norm based fuzzy clustering, *Fuzzy Sets and Systems* 39 (1991) 43–50.
- 650 [33] P. J. Groenen, K. Jajuga, Fuzzy clustering with squared minkowski distances, *Fuzzy Sets and Systems* 120 (2001) 227–237.
- [34] T. Runkler, J. Bezdek, Alternating cluster estimation: a new tool for clustering and function approximation, *IEEE Transactions on Fuzzy Systems* 7 (1999) 377–393.
- 655 [35] H. Frigui, R. Krishnapuram, A robust algorithm for automatic extraction of an unknown number of clusters from noisy data, *Pattern Recognition Letters* 17 (1996) 1223–1232.
- [36] R. Dave, R. Krishnapuram, Robust clustering methods: a unified view, *IEEE Transactions on Fuzzy Systems* 5 (1997) 270–293.
- 660 [37] M.-J. Lesot, B. Bouchon-Meunier, Descriptive concept extraction with exceptions by hybrid clustering, in: *2004 IEEE International Conference on Fuzzy Systems*, volume 1, 2004, pp. 389–394.
- [38] S. Chawla, A. Gionis, k -means-: A unified approach to clustering and outlier detection, in: *Proceedings of the 2013 SIAM International Conference on Data Mining (SDM)*, 2013, pp. 189–197.
- 665 [39] H. Liu, J. Li, Y. Wu, Y. Fu, Clustering with outlier removal, *IEEE Transactions on Knowledge and Data Engineering* 33 (2021) 2369–2379.

- 670 [40] S. Rayana, Odds library, 2016. URL: <http://odds.cs.stonybrook.edu>.
- [41] L. Parsons, E. Haque, H. Liu, Subspace clustering for high dimensional data: A review, SIGKDD Explor. Newsl. 6 (2004) 90–105.