Using Random Forest Distances for Outlier Detection

Antonella Mensi(✉), Ferdinando Cicalese, and Manuele Bicego

Department of Computer Science, University of Verona, Verona, Italy
{antonella.mensi,ferdinando.cicalese,manuele.bicego}@univr.it

Abstract. In recent years, a great variety of outlier detectors have been proposed in the literature, many of which are based on pairwise distances or derived concepts. However, in such methods, most of the efforts have been devoted to the outlier detection mechanisms, not paying attention to the distance measure – in most cases the basic Euclidean distance is used. Instead, in the clustering field, data-dependent measures have shown to be very useful, especially those based on Random Forests: actually, Random Forests are partitioners of the space able to naturally encode the relation between two objects. In the outlier detection field, these informative distances have received scarce attention. This manuscript is aimed at filling this gap, studying the suitability of these measures in the identification of outliers. In our scheme, we build an unsupervised Random Forest model, from which we extract pairwise distances; these distances are then input to an outlier detector. In particular, we study the impact of several Random Forest-based distances, including advanced and recent ones, on different outlier detectors. We evaluate thoroughly our methodology on nine benchmark datasets for outlier detection, focusing on different aspects of the pipeline, such as the parametrization of the forest, the type of distance-based outlier detector, and most importantly, the impact of the adopted distance.

Keywords: Outlier detection · Random forest distances · Data-dependent distances

1 Introduction

Outlier detection is the task of finding abnormal objects in a dataset [12]. These abnormal objects, called outliers, are often considered to be *dissimilar* from the remainder of the data, the inliers. In the literature, many approaches are based on this principle, and solve the outlier detection problem by finding those objects that are distant from the rest of the data, according to a predefined distance [6,17,20]. While many efforts have been put on the derivation of the distance-based method, there is a lack of studies focusing on the proper choice of the distance measure: often the Euclidean distance or other geometric distances are used. These distances may not always be the most suitable choice for an accurate identification of outliers: actually, even though very intuitive from a mathematical perspective, their intrinsic nature also hinders some problems [2] which

S. Sclaroff et al. (Eds.): ICIAP 2022, LNCS 13233, pp. 75–86, 2022.
https://doi.org/10.1007/978-3-031-06433-3_7

may lead to an inaccurate representation of the relation between two objects. An exemplar problem of geometric distances is that they do not consider how data are distributed, i.e., the context: the only factor taken into account are the objects themselves. Another issue is linked to high-dimensionality and data sparsity: often, in such cases, several pairs of objects are equally similar, according to a geometric distance, which may be severely inaccurate. Several studies [1,2,4,9,14,19,21–23] have instead tried to propose a more inclusive definition of similarity. The core concept of all these studies is that similarities should be *data-dependent*, i.e., they should take into account also the context. An example is to consider the density of the space where the objects are, either implicitly or explicitly: given two pairs of objects which are equally similar according to the Euclidean distance, if the first pair is surrounded by fewer objects than the latter, then the former pair of objects should have a higher similarity.

Among the different data-dependent similarity measures that have been proposed in literature, a relevant class is represented by *Random Forests (RF) distances* [4,9,19,21,23], i.e., distances which exploit Random Forests [5] flexibility in describing data. More in detail, a tree in a RF contains several binary tests, each one partitioning the data based on the answer to the test. A RF implicitly encodes the relationship between two objects based on how they answer to the tests, i.e., based on which nodes of the tree the two objects traverse. In general, if the paths of two objects in a tree are highly similar it means that they are near in the space of the problem, since they answer in the same way to the different questions encountered along the path. These characteristics make RFs a valid and flexible distance extractor tool, as confirmed by the variety of successful clustering methodologies that compute the clusters starting from the pairwise distances extracted from a RF [4,13,19,21,23].

Since, as previously mentioned, many outlier-based methodologies work with distances to detect outliers, it would be interesting to investigate the use of refined and informative measures, such as RF-distances, and this represents the main goal of this paper. In our approach, we build a RF based on the extreme version of the *Extremely Randomized Tree* (ERT) structure proposed in [10]: this permits to have an unsupervised model, suitable for the outlier detection task. Once trained the forest, pairwise data-dependent dissimilarities between objects are computed and input to an outlier detector that works with distances. Please note that there exist two very preliminary studies that apply this pipeline [18,21], however they have some limitations: first, they both focus only on one outlier detector method; further, they employ only old RF-distance measures, disregarding recent advances in the field [2,4,21] focuses on only one measure. The scope of the present study is instead much wider: we use 6 RF-distances [2,4,19,21,23], including very recent ones [2,4], making a fairly extended and thorough experimental analysis involving many different outlier detection methods. In detail, the evaluation is made on 9 benchmark outlier detection datasets and we test and compare the extracted RF-distances on 7 outlier detectors; results confirm a general robustness of the approach with an improved detection when using more advanced measures.

The paper is organized into four sections, including the current one. In the following section, Sect. 2, we present in detail the proposed methodology, which is divided into three steps, each thoroughly described. In Sect. 3 we make a thorough experimental evaluation, and lastly in Sect. 4 we make some conclusions.

2 Methodology

This section presents the proposed technique, which can be divided into three steps:

1. Train an ERF $\mathcal{F}$ on a vectorial dataset.
2. From the trained $\mathcal{F}$ extract a dissimilarity matrix $\mathbf{D}$. The entry (i, j) of the matrix contains the pairwise distance between the i^{th} and j^{th} object of the dataset.
3. Input $\mathbf{D}$ to any outlier detector that works with distances and then classify the objects of the dataset as inliers and outliers.

In the next three subsections, we describe each step in detail.

2.1 Step 1: Building an ERF

Formally, an Extremely Randomized Forest $\mathcal{F}$ is composed by T Extremely Randomized Trees [10]. This tree structure is characterized by a high degree of randomness in the building procedure: in its extreme version, called *Totally Randomized Trees*, there is no optimization procedure, and the test of each node is defined completely at random. Since in outlier detection we do not have labels, here we adopt this variant: in detail, first we randomly select a feature, and then we randomly pick a cut-point in the domain of the chosen feature. Please note that in [3] several unsupervised learning strategies for RF have been proposed, aimed at extracting distances for clustering purposes: they show that building trees via completely random splits is beneficial, thus supporting our choice of adopting Totally Randomized Trees.

In detail, each ERT t is built independently of the other trees on a subsample of size S of the training set drawn randomly without replacement. The tree building procedure is recursive: at every node the training objects arrived there are split into two groups according to the chosen random test; in detail, one group follows the left branch and the other follows the right branch. We start from the root with all training objects, and end the splitting procedure when a stopping criterion is met. In particular, a node is labelled as leaf if its depth is greater than a pre-established maximum depth D.

2.2 Step 2: Extracting the Distance Matrix D

The second step consists in the computation of the matrix $\mathbf{D}$: given $\mathcal{F}$ we make all pairs of objects in the dataset traverse an ERT t and compute their distance; the procedure is repeated for all trees; finally the tree distances are aggregated at forest level. We compute **D** using six different proposals of RF-distances [1,2,4,19,21,23] which have shown to be successful in the clustering scenario. In the following, we briefly recall the main principles behind each RF-distance:

- **Shi** [19] defines two objects to be similar only if they end up in the same leaf, i.e., they traverse the same path in a tree. Therefore, at tree level, *Shi* is a binary similarity measure, aggregated at forest level via the average.
- **Zhu2** [23] is a generalization of *Shi* and it assumes that also objects that partially share their paths are similar. The degree of similarity is related to the length of the common path, i.e., to the number of nodes that a pair of objects traverse together before they go separate ways: objects which share a greater portion of their paths are more similar.
- **Zhu3** [23] is a weighted version of *Zhu2*, which considers that different nodes in the tree convey different information; in particular, each node is given a weight which is inversely proportional to the number of training objects that reached such node (nodes with few training objects describe the space in a more refined way).
- **RatioRF** [4] implements the Tversky's ratio model of similarity [22] in the context of RF. *RatioRF* computes the similarity between two objects by considering all nodes in the tree that are in the traversed path of at least one of the two objects, and not only nodes that are in both paths. In detail, the similarity between two objects increases if, given an internal node present only in one path, the two objects would traverse the same edge.
- **Ting** [21] implements in the context of RF the m_p*–dissimilarity*, a mass-based distance function proposed by [1]. The rationale is that the similarity between two objects is inversely proportional to the number of training objects contained in the minimum region enclosing both objects. In case of multidimensional data, the minimum enclosing region, and therefore the similarity, is defined independently for each dimension and then results are aggregated via the arithmetic mean. In the context of RF, as defined by [21], the m_p*–dissimilarity* corresponds to the number of training objects contained in the Lowest Common Ancestor (LCA) of the two objects since it represents the last node containing both of them. The aggregation at forest level is computed via the arithmetic mean, as defined by [1].
- **Aryal** [4] implements the mass-based distance function proposed by [2], the m_0*-dissimilarity.* At tree level, it is equivalent to *Ting*, but the aggregation at forest level is computed using the geometric mean. Using the multiplication increases the impact of those trees in which the similarity between two objects is either very high or very low.

2.3 Step 3: Distance-Based Outlier Detection

The extracted distance matrix **D** contains the pairwise distances between each pair of objects in a dataset, to be fed to a distance-based outlier detection methodology. A great variety of methodologies that work with distances have been designed; in the following, we briefly describe the main principles behind those we employed in our proposal:

- **Nearest Neighbor (NN)-based techniques** [20]**.** This represents a group of very simple yet often well performing methodologies. We employed four different variants: i) **KNNd** evaluates the ratio of the distance between an object and its K^{th} nearest neighbor to the distance between the latter and its K^{th} nearest neighbor. If the former distance is much bigger than the latter, then the object under analysis has an increased probability of being an outlier; ii) **KNNDist** is simply the distance to the K^{th} nearest neighbor: it assumes that an outlier will be distant from its own K^{th} nearest neighbor independently of the density of the latter; iii) **KNNd-Av** is analogous to *KNNd* but instead of considering just the K^{th} nearest neighbor it takes into account all of them via computing the average; and iv) **KNNDist-Av** is the average distance of the first K^{th} neighbors. The main drawback of all these methods is setting K.
- **Local Outlier Factor (LOF)** [6]**.** This represents a more complex technique based on the estimation of the relative density, computed using the distances. In particular, *LOF* compares the density of $\mathbf{x}$'s neighborhood to that of each of its neighbors. If there is at least one neighbor which has a much denser neighborhood, then the probability of $\mathbf{x}$ being an outlier increases, since the difference in density is a hint that $\mathbf{x}$ is distributed quite differently from at least one of its neighbors. Analogously to NN-based techniques, there is the drawback of setting the size of the neighborhood K.
- **K-Centers** [20]**.** This is a clustering-based technique, which, after computing the clusters, calculates the distance of each object to the center of the cluster to which it belongs: if an object is far away from said center then it is more likely to be an outlier. The main problem of *K-Centers* consists of setting an appropriate number of clusters K.
- **ProxIF** [17]**.** This represents a RF-based methodology for outlier detection that works with pairwise distances, being inspired by the well known Isolation Forest [15]. An object is more likely to be an outlier if, on average, it traverses a shorter path in the forest. Unlike the other detectors, there exists a default parametrization to build the forest which works well in most cases – the parameters to set consists of a training criterion, and S, T and D (the same ones of an ERF).

3 Experimental Evaluation

This section is dedicated to the experimental evaluation of the methodology. In the first subsection, Subsect. 3.1, we describe the datasets and other experimental details. Subsequently, in Subsect. 3.2 we study the impact of the ERF parametrization, whereas in Subsect. 3.3 we assess whether there is a best outlier detector to work with. Lastly, in Subsect. 3.4 we present several analyses concerning the behaviour of different distance measures.

3.1 Experimental Details

We perform the evaluation of the methodology on 9 different benchmark datasets for outlier detection [7,11,15,16], 6 of which are UCI-ML datasets[1] preprocessed according to [11]. Instead, *Cardiotocography, Hepatitis* and *Stamps* were taken from [7] and preprocessed accordingly. In Table 1, we report the number of objects and features and the percentage of outliers for each dataset. We can observe that the datasets cover a large range of cases, differing greatly in dimensionality (from 5 up to 164), in the outlier percentage (from 5.39% up to 45.80%) and in the number of samples (the smallest one having 80 objects whereas the biggest 7200). Many different experiments were performed by varying the value of several parameters:

- Number of trees T: $50, 100, 200$. 3 options.
- Number of samples used to train each tree S: $64, 128, 256$. 3 options.
- Maximum depth D each tree can reach: $\log_2(S), S - 1$. 2 options.
- Distance measure: *Shi, Zhu2, Zhu3, RF-Ratio, Aryal, Ting*. 6 options.
- Outlier Detector: *KNNd*, *KNNDist*, *KNNd-Av*, *KNNDist-Av*, *K-Centers*, *LOF*, *ProxIF*. 7 options.

Each experiment, i.e., parametrization setting, was iterated 10 times. The iterations have been created by splitting randomly the dataset in half, one half for the training set and the other for the testing set, with the only constraint related to the absence of outliers in the former. In addition, given a dataset, the 10 partitions of training and testing sets are identical across all parametrizations. As usually done in the outlier detection field, all experiments have been evaluated by measuring the performance in terms of Area Under the ROC Curve (AUC). Further, for all statistical analyses, we set the significance level to $\alpha = 0.05$.

3.2 ERF Parametrization

The first analysis aims at finding the best ERF parametrization for each distance measure, to be used in the subsequent analyses. Indeed, different distance measures may benefit from a different parametrization setting: for example, as to the depth D, whereas some distances may suffer, others may benefit from the additional information contained in deeper trees. We made three different analyses, one for each parameter (T, S, and D): however, due to a lack of space, we report here only the analysis concerning the maximum reachable depth D, being the most interesting one, whereas for S and T we simply summarize the obtained results at the end of the section. For what concerns the analysis of the depth, given a value of D and a distance measure, we compute the average AUC for each dataset and for every parametrization (obtained by varying S and T) across the related iterations – we used as outlier detector only *NNd* (analogous to *KNNd* with $K = 1$), one of the simplest techniques for which no parameter must be set. Then, for each distance measure, we perform a Wilcoxon signed-rank test to compare the two sets of results for the two values of D. We depict

[1] Available at https://archive.ics.uci.edu/ml/index.php.

Table 1. Overview of the 9 datasets used for the experimental evaluation.

Datasets	Nr. of objects	Nr. of features	Outlier %
Annthyroid	7200	6	7.42%
Arrhythmia	452	164	45.80%
Cardiotocography	2126	21	22.15%
Hepatitis	80	19	16.25%
Ionosphere	351	32	35.90%
Pima	768	8	34.90%
Spambase	4601	57	39.40%
Stamps	340	9	9.12%
Wilt	4839	5	5.39%

Table 2. Statistical analysis for D.

Distance	Rank		p-value
	$D = \log_2(S)$	$D = S - 1$	
Shi	1.28	1.72	**1.7e−5**
Zhu2	1.21	1.79	**2.76e−10**
Zhu3	1.26	1.74	**1.36e−7**
Ting	1.62	1.38	**0.014**
RatioRF	1.52	1.48	0.701
Aryal	1.63	1.37	**6.04e−4**

the results of such analysis in Table 2: we report for each distance measure the mean rank for each value of D and the p-value output by the test. We highlight in **bold** the p-value if the difference between $D = \log_2(S)$ and $D = S - 1$ is statistically significant. From the table different observations can be derived: first, we can infer that it is significantly better to extract *Shi, Zhu2* and *Zhu3* from trees built until $D = \log_2(S)$ rather than from trees built to the end. Indeed, the probability of two objects ending up in the same leaf (and in general of having a longer common path) decreases as D increases. In other words, if we extract *Shi* from a tree where $D = S - 1$ we may obtain a sparse similarity matrix and probably less discriminative in the detection of outliers[2]. A second observation we can infer from Table 2 is that *RatioRF* is independent of the used D, i.e., it is very robust independently of the tree structure; in the subsequent analyses we therefore set $D = \log_2(S)$ for *RatioRF* – smaller trees are less expensive from a computational point of view. As to *Ting* and *Aryal*, the two mass-based RF-distances, we observe an opposite behaviour with respect to *Shi*, *Zhu2* and

[2] Please note that we can extend this reasoning to *Zhu2* and *Zhu3*: the resulting matrix may not be sparse, but it may contain many low similar values, thus impacting on the final outlier detection step.

Zhu3: $D = S - 1$ is ranked higher than $D = \log_2(S)$ and the difference is statistically significant. Indeed, it is likely that when growing trees to their maximum depth, we have an increased variability in these kinds of distance matrices. In other words, in trees where $D = \log_2(S)$ there may be several pairs of objects ending up in the same LCA and therefore getting the same distance value even though their *true* distance may be different.

In conclusion, we set $D = \log_2(S)$ for all RF-distances except *Aryal* and *Ting*, for which we set $D = S - 1$. As to the other parameters of an ERF, S and T, from the related analyses, we discovered a common behaviour among the different distances: in all cases the most suitable choice is to set $S = 64$ and $T = 50$.

3.3 Comparison of Outlier Detectors

In this section we assess whether, independently of the used RF-distance measure, there is a best outlier detector to use. First, after a thorough preliminary analysis not shown here, we set the parameters of each outlier detector as follows: *KNNd*, $K = 8$; *KNNDist*, $K = 4$; *KNNd-Av*, $K = 16$; *KNNDist-Av*, $K = 8$; *K-Centers*, $K = 5$; *LOF*, $K = 9$. As for *ProxIF* we set the parameters according to the guidelines presented in [17]: $T = 200, S = 256, D = \log_2(S)$ and $O-2PS_D$ as training criterion. Then we compute for each outlier detector the AUC values for each dataset and distance measure, averaging them across the 10 iterations. Subsequently, we make a non-parametric statistical analysis to compare the seven outlier detectors. In detail, we carried out a Friedman test followed by a post-hoc Nemenyi test: the former is needed to uncover whether there is a global statistically significant difference among all models, whereas the latter is used to discover the pairs of statistically different outlier detectors. In Fig. 1 the results of the statistical analysis are visualized as a critical difference (CD) diagram [8]: each outlier detector is represented via its rank on a single line, with the best rank represented on the right. Whenever two (or more) models are comparable, i.e., there is no significant difference, they are connected by a line. From Fig. 1 we can observe that the first-ranked outlier detector is *KNNDist-Av*; nevertheless, its performances are comparable to other three detectors in the ranking, confirming the robustness of the methodology across several outlier detectors based on different core concepts.

3.4 Comparison of Distance Measures

In this section we make three analyses: we compare the distances on the best outlier detector, then we try to infer whether some RF-distances are more suitable than others, and lastly we assess whether there is a best combination of distance measure and outlier detector.

The first analysis compares the performances of the 6 distance measures when using *KNNDist-Av*, the best outlier detector for this task according to the analysis made in the previous section. In Table 3, given a dataset and a distance measure, we report the average AUC across the iterations and indicate

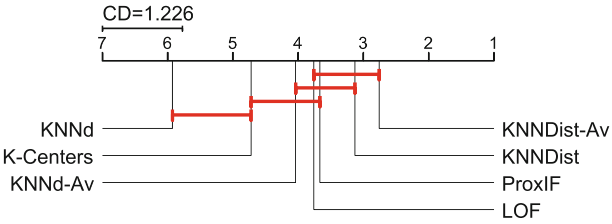

Fig. 1. Comparing outlier detectors independently of the distance measure via a CD diagram.

Table 3. Evaluation of the 6 RF-distances using KNNDist-Av.

Dataset	Shi	Zhu2	Zhu3	RatioRF	Ting	Aryal
Annthyroid	0.8560	0.8861	0.8647	0.8854	0.9044	**0.9147**
Arrhythmia	0.8131	0.8137	0.8129	**0.8210**	0.8179	0.7880
Cardiotocography	0.4564	0.4716	0.4654	**0.4935**	0.4116	0.4421
Hepatitis	0.6815	0.6989	0.6758	0.6906	0.7271	**0.7863**
Ionosphere	0.9492	0.9521	0.9466	0.9415	0.9701	**0.9786**
Pima	0.7189	0.7279	0.7184	0.7255	0.7389	**0.7538**
Spambase	0.8550	0.8589	0.8517	0.8580	0.8690	**0.8800**
Stamps	0.9236	0.9362	0.9310	**0.9395**	0.8525	0.7634
Wilt	0.7131	0.6735	0.6805	0.6827	0.7928	**0.8296**

Table 4. Best RF-distances for each outlier detector.

Outlier detector	Best gistances			
KNNDist-Av	Aryal	Ting	RatioRF	
KNNd	RatioRF	Aryal	Ting	Zhu2
KNNDist	Aryal	Zhu2	RatioRF	Ting
KNNd-Av	RatioRF	Zhu2	Aryal	Ting
K-Centers	RatioRF	Zhu2	Shi	Zhu3
LOF	RatioRF	Aryal	Zhu2	
ProxIF	Aryal	Ting		

in **bold** the best result for each dataset. We can observe that all RF-distances seem to have good and consistent performances across all datasets. The only exceptions are: *Wilt* for which using a non-mass-based distance measure leads to much poorer performances and *Stamps* for which the converse holds, i.e., using either *Aryal* or *Ting* relevantly decreases the performances. From Table 3 we can also observe that for each dataset, the best distance measure is either *Aryal* or *RatioRF*.

The second analysis compares the performances of the 6 distance measures on each outlier detector. In detail, starting from the results obtained on each dataset, we carried out a Friedman test followed by a post-hoc Nemenyi test. In Table 4 we report for each detector the best distance measures, i.e., those that were either first in the ranking or comparable to the first-ranked measure according to the statistical analyses we carried out. We report them left to right, from best to worst ranked. We can observe that as to *KNNDist-Av*, the best outlier detector for this task, *Aryal* is the best choice, even though it is comparable to both *RatioRF* and *Ting*. Overall, we can observe that *Aryal* and *RatioRF* are the only distance measures that are either the first-ranked or comparable to such measure for six out of seven outlier detectors. From Table 4 we can also conclude that using *Shi* and *Zhu3* is overall a bad choice.

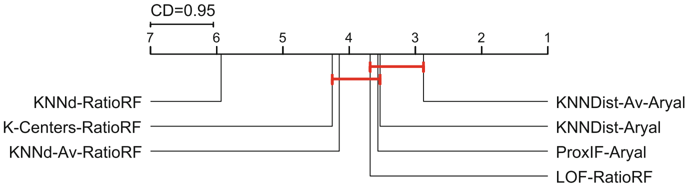

Fig. 2. CD diagram that compares each classifier combined with its best distance measure.

The last analysis consists of comparing the first-ranked distances for each outlier detector via the Friedman and Nemenyi tests, which results are depicted in the CD diagram in Fig. 2. The analysis confirms that *KNNDist-Av* seems to be the best outlier detector even though it is comparable to *KNNDist*, *LOF* and *ProxIF*, analogously to the observation made in the previous section. Three of the best ranked detectors work with the mass-based RF-distance *Aryal* confirming its suitability in detecting outliers.

Summarizing the observations made in this section, there are two distance measures which seem to be the most suitable for outlier detection: **Aryal** and **RatioRF**. Nevertheless, it is difficult to establish which between the two is the best one, since their performance also depends on the dataset and on the outlier detector.

4 Conclusions

In this paper we presented a thorough study on using RF-distances, i.e., informative data-dependent measures extracted from a RF model, to detect outliers. Indeed, even though there exist several outlier detectors based on distances, there is a lack of research on distance measures. Our manuscript, with respect to previous works, thoroughly studies the impact of several RF-distances, including quite recent and refined ones, on the identification of outliers using different outlier detectors.

The suitability of this contribution has been tested on a total of 9 datasets. It is shown how more refined distance measures, such as Aryal and RatioRF, are overall a better choice than more simplistic ones, such as Shi. Further, even though the best results are reached with a rather simple NN-based outlier detector, the methodology has shown to be robust across several classifiers based on different principles.

References

1. Aryal, S., Ting, K.M., Washio, T., Haffari, G.: Data-dependent dissimilarity measure: an effective alternative to geometric distance measures. Knowl. Inf. Syst. **53**(2), 479–506 (2017). https://doi.org/10.1007/s10115-017-1046-0
2. Aryal, S., Ting, K.M., Washio, T., Haffari, G.: A comparative study of data-dependent approaches without learning in measuring similarities of data objects. Data Min. Knowl. Discov. **34**(1), 124–162 (2019). https://doi.org/10.1007/s10618-019-00660-0
3. Bicego, M., Escolano, F.: On learning random forests for random forest clustering. In: Proceedings of the International Conference on Pattern Recognition, pp. 3451–3458 (2020). https://doi.org/10.1109/ICPR48806.2021.9412014
4. Bicego, M., Cicalese, F., Mensi, A.: RatioRF: a novel measure for random forest clustering based on the Tversky's ratio model. IEEE Trans. Knowl. Data Eng. 1 (2021). https://doi.org/10.1109/TKDE.2021.3086147
5. Breiman, L.: Random forests. Mach. Learn. **45**, 5–32 (2001). https://doi.org/10.1023/A:1010933404324
6. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: LOF: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, pp. 93–104 (2000). https://doi.org/10.1145/335191.335388
7. Campos, G.O., et al.: On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. Data Min. Knowl. Discov. **30**(4), 891–927 (2016). https://doi.org/10.1007/s10618-015-0444-8
8. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. **7**, 1–30 (2006)
9. Englund, C., Verikas, A.: A novel approach to estimate proximity in a random forest: an exploratory study. Expert Syst. Appl. **39**(17), 13046–13050 (2012). https://doi.org/10.1016/j.eswa.2012.05.094
10. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. Mach. Learn. **63**(1), 3–42 (2006). https://doi.org/10.1007/s10994-006-6226-1
11. Goix, N., Drougard, N., Brault, R., Chiapino, M.: One class splitting criteria for random forests. In: Zhang, M.L., Noh, Y.K. (eds.) Proceedings of 9th Asian Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 77, pp. 343–358 (2017)
12. Hawkins, D.M.: Identification of Outliers, vol. 11. Springer, Dordrecht (1980). https://doi.org/10.1007/978-94-015-3994-4
13. Kaufman, L., Rousseeuw, P.J.: Finding Groups in Data: An Introduction to Cluster Analysis, vol. 344. Wiley, Hoboken (2009). https://doi.org/10.1002/9780470316801
14. Lin, D.: An information-theoretic definition of similarity. In: Proceedings of the International Conference on Machine Learning, vol. 98, pp. 296–304 (1998)

15. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: IEEE International Conference on Data Mining, pp. 413–422 (2008). https://doi.org/10.1109/ICDM.2008.17
16. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation-based anomaly detection. ACM Trans. Knowl. Discov. Data **6**(1), 3:1–3:39 (2012). https://doi.org/10.1145/2133360.2133363
17. Mensi, A., Bicego, M., Tax, D.M.: Proximity isolation forests. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 8021–8028. IEEE (2021). https://doi.org/10.1109/ICPR48806.2021.9412322
18. Mensi, A., Franzoni, A., Tax, D.M.J., Bicego, M.: An alternative exploitation of isolation forests for outlier detection. In: Torsello, A., Rossi, L., Pelillo, M., Biggio, B., Robles-Kelly, A. (eds.) S+SSPR 2021. LNCS, vol. 12644, pp. 34–44. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-73973-7_4
19. Shi, T., Horvath, S.: Unsupervised learning with random forest predictors. J. Comput. Graph. Stat. **15** (2005). https://doi.org/10.1002/sam.11498
20. Tax, D.: One-class classification; concept-learning in the absence of counter-examples. Ph.D. thesis, Delft University of Technology (2001)
21. Ting, K., Zhu, Y., Carman, M., Zhu, Y., Zhou, Z.H.: Overcoming key weaknesses of distance-based neighbourhood methods using a data dependent dissimilarity measure. In: Proceedings of the International Conference on Knowledge Discovery and Data Mining, pp. 1205–1214 (2016). https://doi.org/10.1145/2939672.2939779
22. Tversky, A.: Features of similarity. Psychol. Rev. **84**(4), 327 (1977). https://doi.org/10.1037/0033-295X.84.4.327
23. Zhu, X., Loy, C., Gong, S.: Constructing robust affinity graphs for spectral clustering. In: Proceedings of the International Conference on Computer Vision and Pattern Recognition, pp. 1450–1457 (2014). https://doi.org/10.1109/CVPR.2014.188