Anomaly Detection via Neighbourhood Contrast

Bo Chen$^1$, Kai Ming Ting$^2$, and Tat-Jun Chin$^1$

$^1$ The University of Adelaide, Adelaide, South Australia 5005, Australia
$^2$ National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China

Abstract. Relative scores such as Local Outlying Factor and mass ratio have been shown to be better scores than global scores in detecting anomalies. While this is true, our analysis reveals for the first time that these relative scores have a key shortcoming: anomalies have greatly different relative scores if they are located in different regions where the curvatures of the density surface are very different. As a result, the low-score anomalies could be ranked lower than some normal points. This revelation motivates (i) a new score called Neighbourhood Contrast (NC) which produces approximately the same high scores for all anomalies, regardless of varying curvatures of the density surface in different regions; and (ii) an anomaly detection method based on NC. Our experiments show that the proposed method which employs the new score significantly outperforms methods using the aforementioned relative scores on benchmark datasets.

Keywords: Neighbourhood Contrast · Anomaly detection · Relative scores

1 Introduction

Anomaly detection is an important task that has been widely used in many applications such as intrusion detection [20,21], clinical diagnosis [14,15,24,27], outlier image removal [18], activity recognition [11] and power grid event detection [26]. One way to detect anomalies is using density-based scores. However, naively using densities to indicate data points’ outlyingness does not generate satisfactory results. To detect both global and local anomalies, researchers have advocated the use of relative scores. For example, Local Outlying Factor (LOF) [5] computes a local reachability density ratio for each point based on a k-nearest-neighbour density estimator; ReMass-iForest (RMF) [3] calculates the ratio of the numbers of points that fall into a leaf node and its immediate parent node in an isolation tree [16]. Local Outlier Probabilities (LoOP) employs the inverse ratio of probabilistic set distances [13], which can be viewed as a variant of density ratio. These relative scores have been shown to produce better anomaly detection performance [3,5] than their global counterparts such as $k$-th nearest neighbour distance [1,4,22] and path length [16,17].
Despite their popularity, our investigation reveals a key shortcoming of these ratio-based relative scores which has not been identified previously. Existing ratio-based relative scores are designed to be sensitive to the local data distribution. However, we discover that such relative scores are affected by not only local densities, but also the curvature (i.e., the first and second order derivatives) of the density surface, which can yield undesirable outcomes.

Figure 1 presents such an example, where anomalies A and B are located at low and high curvature surfaces, respectively. Existing scores RMF and LOF produce low score for A and high score for B, leading A to be ranked lower than some normal points. In other words, if two local density minima have the same density values, the one with a higher curvature will have a larger score. As a result, anomalies at locations of low curvature are harder to detect.

Recent research has proposed a notion of Neighbourhood Contrast (NC) [6] as a substitute of density to detect clusters in clustering applications. In this paper we propose a new score based on the concept of NC, which is sensitive to the contrast between two nearby locations of a local region while at the same time being robust to varying curvatures of the density surface in different regions of the data space. Figure 1(d) demonstrates such characteristic: the proposed NC produces the highest score for both anomalies.

We make the following contributions in this paper:

- Identifying a key shortcoming of existing ratio-based relative anomaly scores.
- Analysing the condition under which these scores fail to rank anomalies properly.
- Proposing a new anomaly score called Neighbourhood Contrast which does not have the identified shortcoming.
- Creating a new anomaly detection method called Neighbourhood Contrast Anomaly Detector (NCAD) which outperforms LOF, LoOP and two existing tree-based methods iForest [16] and RMF. The new method is also more scalable than LOF and LoOP to large datasets.
2 A Key Shortcoming of Relative Scores

A relative score is often defined as the aggregation of the ratios between the measurements of the query point and its neighbourhood, such as LOF [5], LoOP [13] and RMF [3]. A general form of relative score $\tau(x)$ can be defined as:

$$\tau(x) = A_{y \in R(x)} \left( \frac{f(y)}{f(x)} \right),$$

where $f$ is a non-negative base measure such as density, inverse of distance or mass; $A$ is an aggregation operation such as sum, average, or expectation; and $R(x)$ is the neighbourhood area of $x$.

We are interested in the behavior of such a relative score particularly at local density minima, because they are where global and local anomalies are located. Let $A$ be the average operation, $f(x)$ be a 1-dimensional parabolic curve, $x_0$ be a local minimum of $f(x)$ and $R(x) = [x_0 - \Delta, x_0 + \Delta]$. By applying the Taylor expansion up to the second order, the relative score can be derived as

$$\tau(x_0) = \frac{1}{2\Delta} \int_{x_0-\Delta}^{x_0+\Delta} \frac{f(x)}{f(x_0)} dx$$

$$= 1 + \frac{f''(x_0)}{4\Delta f(x_0)} \int_{x_0-\Delta}^{x_0+\Delta} (x - x_0)^2 dx$$

$$= 1 + c \frac{f''(x_0)}{f(x_0)},$$

(1)

where $c \geq 0$ only depends on $R$. Since $x_0$ is the local minimum, the curvature at $x_0$ determines the second order derivative $f''(x_0)$. That is, if all else being equal (i.e. the neighbourhood $R$ is fixed, the local minima have the same value), the larger the curvature, the larger the relative score. This is illustrated in Fig. 2. By Eq. (1), $\tau_f(x_0) < \tau_g(x_0)$ because $f''(x_0) < g''(x_0)$.

In a nutshell, ignoring the additive constant 1, the relative score of a local minimum point is proportional to the curvature or the second derivative of the function $f$, and inversely proportional to $f$, where $f$ can be either density, mass or any other base measure.

Figure 1 provides an example of aforementioned phenomenon with a synthetic dataset where two anomalies are at locations of different curvatures. Both existing scores LOF and RMF, shown in Figs. 1(b) and (c), exhibit the behaviour as predicted by Eq. (1): the anomalies have notably different relative scores because of the different curvatures at the two locations. This lowers the ranking of the low-curvature point and makes it harder to be detected.

In summary, the ratio-based relative scores are sensitive to not only the changes of local densities, but also the second derivatives of the density surface. A desirable score should produce large scores for all anomalies while being robust to varying density curvatures. To avoid the sensitivity to curvature, one can use a score based on contrast rather than ratio as follows:

$$\tau(x_0) = \frac{1}{2\Delta} \int_{x_0-\Delta}^{x_0+\Delta} I(f(x) > f(x_0)) dx$$

(2)
650 B. Chen et al.

Fig. 2. $f$ and $g$ are functions with different curvatures and $f(x_0) = g(x_0)$, $f'(x_0) = g'(x_0) = 0$; $R$ is the same.

where $I(\cdot)$ is an indicator function returning 1 if the condition is met, and 0 otherwise. If $x_0$ is a local minimum, Eq. (2) equals to 1, regardless of the curvature at $x_0$.

Figure 1(d) shows that the proposed new score Neighbourhood Contrast (NC) is such a score—it yields the highest score for these two anomalies in comparison with all other points. This figure is produced from NCAD, an algorithm which employs NC. We describe NC and NCAD in the next two sections.

3 Neighbourhood Contrast: A New Score

We propose a new anomaly score based on the concept of Neighbourhood Contrast (NC). The notion of NC [6] was first proposed as an alternative to density in clustering because of its robustness to density variation. We find that the characteristics of NC are also desirable in detecting local anomalies. Here we reiterate the definition of NC with key modifications to make it suitable to be an anomaly score.

For a point $x \in \mathbb{R}^d$, let $T$ be a random process which generates pairs of non-overlapping and equal size regions in $\mathbb{R}^d$; let $T(x)$ and $T'(x)$ be the pair of regions that is generated by $T$ with respect to $x$. More specifically, $T(x)$ denotes the region in the pair that contains $x$ and $T'(x)$ denotes the other region not containing $x$. Given a dataset $D$ and a region $R$, let $D(R) := \{x \in D : x \in R\}$ denote the set of data points in $D$ that are covered in the region $R$.

Definition 1. Given a dataset $D$ and $T$, the Neighbourhood Contrast of a point $x$ is the probability that $T(x)$ has smaller mass than $T'(x)$, i.e.,

$$NC(x) = P(|D(T(x))| < |D(T'(x))|),$$

where $|\cdot|$ measures the cardinality of a set.

Generally $T$ can be any process that generates pairs of regions as described above. Although other implementation is possible, in this paper we only focus on
one implementation of $T$ which is a binary tree that partitions the data space. Consequently, $T(x)$ and $T'(x)$ denote the leaf node of tree $T$ that contains $x$ and its sister node, respectively. For convenience, in the rest of this paper we use notations like $T(x)$ to denote the node of a tree, its corresponding region in the data space, and the set of training data points that fall in this node/region. The exact meaning should be easy to tell based on the context. For example, the definition of NC can be simplified as $NC(x) = P(|T(x)| < |T'(x)|)$.

3.1 Property of NC

Theorem 1. If $x^*$ is a local density minimum and the densities of its neighbouring area are isotropically increasing, then $NC(x^*) = 1$.

Proof. Let $x^*$ be an isotropic density minimum, as shown in Fig. 3. For any $x$, the longer the distance $d(x, x^*)$, the higher the density of $x$. Suppose a random pair of areas $T(x^*)$ and $T'(x^*)$ is generated as shown in Figs. 3(b) and 3(c). For any point $x$ in $T(x^*)$ and its mirror counterpart $x'$ in $T'(x^*)$, it holds that $d(x, x^*) < d(x', x^*)$; consequently $f(x) < f(x')$, for all $x \in T(x^*)$. Therefore, we have $\int_{T(x^*)} f(x) dx < \int_{T'(x^*)} f(x') dx'$. In other words, the probability mass in $T(x^*)$ is always smaller than that in $T'(x^*)$, which leads to $NC(x^*) = 1$.

Note that Theorem 1 holds regardless of the first and second order derivatives of the function $f$. Also note that even if the monotonically increasing distribution around $x^*$ is not isotropic in practice, it is still very likely that $T(x^*)$—the area which contains the minimum density point—has lower average density than its neighbouring area $T'(x^*)$; and hence $T(x^*)$ is likely to has a smaller mass than $T'(x^*)$. Therefore, Theorem 1 implies the following property of NC.

Property 1. The minimum density point $x^*$ in a region of monotonically increasing densities has Neighbourhood Contrast $NC(x^*)$ approximates 1, regardless of the densities or the curvature of the density surface in the region.

3.2 Implementation and Estimation of NC

We use random binary trees to generate the neighboring areas $T$ for NC estimations. We call this type of trees NCTrees which has the following features:

- Non-axis-parallel splitting and round-robin attribute selection are applied at each internal node of a tree. This produces a contour map of anomaly score that is not restricted by axis-parallel patterns.
- Height limit determined by a parameter $L$ only: each branch of a tree stops splitting to form a leaf node if the training subset has no more than $L$ points. This allows the tree to have different levels in regions of different densities, producing adaptive leaf node sizes.
- Equal-volume split point: An initial node space is generated in the root node; and each internal node is split into two half spaces of equal-volume. This is required by the definition of NC.
Fig. 3. (a) The density minimum point $x^*$ of an isotropic monotonically increasing function, where concentric contours centered at $x^*$. (b) Example random areas $T(x^*)$ and $T'(x^*)$. (c) An arbitrary point $x$ in $T(x^*)$ and its mirror counterpart $x'$ in $T'(x^*)$: $x'$ is always further away from $x^*$ than $x$.

The tree generation process is summarized as follows. Given a dataset $D \subset \mathbb{R}^d$, an ensemble of $t$ NCTrees $\{T_i\}_{i=1}^t$ is built. Before growing an NCTree $T_i$, a random rotation is applied to $D$. Let $u_i$ be a randomly oriented orthonormal basis. Let $D$ also denote the dataset arranged in a matrix form, we produce $D_i = Du_i$, which is then used to generate an axis-aligned hyper-rectangular region $S_i$ that envelopes $D_i$. The algorithm which generates $S_i$ is given in Algorithm 1.

Algorithm 1: Generate Initial Space($D$)

input : $D$ - dataset;
output: $S$ - an axis-aligned hyper-rectangular region such that $D \subset S$
1 $d \leftarrow Number\_of\_dimension(D)$
2 for $q = 1, \ldots, d$ do
3     $min_q \leftarrow \min\{x_q : x \in D\}$
4     $max_q \leftarrow \max\{x_q : x \in D\}$
5     $z_q \leftarrow$ uniformly random value in $[min_q, max_q]$  
6     $r_q \leftarrow max_q - min_q$
7     $S^l_q \leftarrow z_q - r_q$, the lower bound of $S$ on $q$
8     $S^u_q \leftarrow z_q + r_q$, the upper bound of $S$ on $q$
9 end

The tree building process is given in Algorithm 2. The initial space $S_i$ is used as the root node space of $T_i$. At each level of $T_i$, the same feature $q \in \{1, \ldots, d\}$ is used as the splitting feature for all internal nodes created at this level. Different levels of a tree use different features, selected in a round-robin manner. For each internal node, the node space is split in the middle point of feature $q$, producing two equal-volume half-spaces, which are passed to its child nodes as their respective node spaces. A node becomes a leaf node and stops splitting if its mass, i.e., the number of training points that fall in it, is no greater than a threshold $L$. 
Algorithm 2: Build_NCTree($D, S, q, \mathcal{L}$)

**input:** $D$ - dataset; $S$ - node space; $q$ - current splitting attribute; $\mathcal{L}$ - leaf node mass threshold

**output:** $T$ - a binary tree that partitions $S$

1. if $|D| \leq \mathcal{L}$ then
2. Terminate and return $S$ as a leaf node region
3. else
4. $q \leftarrow q + 1$
5. if $q > d$ then
6. $q \leftarrow q - d$
7. end
8. $s_q \leftarrow (S_q^l + S_q^u)/2$
9. $D_l \leftarrow \{x \in D : x_q < s_q\}$
10. $D_r \leftarrow \{x \in D : x_q \geq s_q\}$
11. Split $S$ at $s_q$ into $S_l$ and $S_r$
12. left $\leftarrow$ Build_NCTree($D_l$, $S_l$, $q$, $\mathcal{L}$)
13. right $\leftarrow$ Build_NCTree($D_r$, $S_r$, $q$, $\mathcal{L}$)
14. end

After an ensemble of trees $\{T_i\}_{i=1}^t$ is built, $NC(x)$ is estimated by

$$NC(x) = \frac{1}{t} \sum_{i=1}^t I(|T_i(x)| < |T_i'(x)|).$$

Note that the tree-building process described above is similar to that in [6]. A key difference is that we remove the upper limit of tree levels, i.e., the $h$ parameter in [6]. By eliminating the $h$ parameter, the tree-building process will be terminated solely by the $\mathcal{L}$ threshold. The reason for getting rid of $h$ is because, in anomaly detection, we do not need to control the size of the maximum level node regions, which is designed for producing better clustering results. In contrast, letting trees grow until reaching the $\mathcal{L}$ threshold allows leaf nodes to better adapt to different densities. That is, dense areas will be more sufficiently partitioned to produce more accurate estimations.

4 Neighbourhood Contrast Anomaly Detector

Here we introduce a new method named Neighbourhood Contrast Anomaly Detector or NCAD. Given a dataset $D$, an ensemble of NCTrees is built to estimate NCs of all points in $D$. Points having the highest NCs are designated as anomalies. The algorithm of NCAD is given in Algorithm 3.

NCAD has only one parameter $\mathcal{L}$ that needs to be tuned; it controls the tree size. We recommend $\mathcal{L}$ be set to a proportion of the given dataset size.
Figure 4 provides an example of NCAD distributions represented with heat maps, generated using different settings of $\mathcal{L}$.

Algorithm 3: NCAD($D, t, h, \mathcal{L}$)

input: $D$ - dataset; $t$ - ensemble size; $\mathcal{L}$ - leaf node mass threshold
output: $NC(x), \forall x \in D$ - anomaly scores of $D$

1 for $i = 1, \ldots, t$ do
2 \hspace{1em} $u_i \leftarrow$ a randomly orientated orthonormal basis of $\mathbb{R}^d$
3 \hspace{1em} $D_i \leftarrow Du_i$
4 \hspace{1em} $q \leftarrow$ a randomly selected value in $\{1, \ldots, d\}$
5 \hspace{1em} $S_i \leftarrow \text{Generate}_\text{Initial}_\text{Space}(D_i)$
6 \hspace{1em} $T_i \leftarrow \text{Build}_\text{NCTree}(D_i, S_i, q, \mathcal{L})$
7 end
8 for $j = 1, \ldots, |D|$ do
9 \hspace{1em} $NC(x_j) = \frac{1}{t} \sum_{i=1}^{t} I(|T_i(x_j)| < |T'_i(x_j)|)$
10 end
11 Return sorted $NC(x), \forall x \in D$ in descending order

![Fig. 4. Heat maps of NCAD with different settings of $\mathcal{L}$.](image)

5 Experiments

We compare the anomaly detection performance among NCAD, iForest [16], LOF [5], RMF [3] and LoOP [13]. A synthetic dataset and 12 benchmark datasets\footnote{The sources of the datasets are: mulcross [23]; smtp [25]; wilt [12]; htru2 [19]; and the rest are from UCI repository [9].} are used in the evaluation.

For all methods, their key parameters are searched in certain ranges and the best AUCs are recorded. The search ranges of parameters are given in Table 1.
For NCAD, iForest and RMF, the ensemble size $t$ is set to 100. We report the average AUCs of NCAD, iForest and RMF over 10 runs, since they are randomized methods. The AUCs of LOF and LoOP are the results of one run only since they are deterministic methods.

**Table 1.** Parameter search values for NCAD, RMF, iForest, LOF and LoOP.

| Method     | Key parameters and search range                      |
|------------|------------------------------------------------------|
| NCAD       | $\mathcal{L}$: \{0.001, 0.005, 0.01, 0.05, 0.1, 0.15, 0.2, 0.3, 0.4, 0.5\} $\times n$ |
| RMF and iForest | $\psi$: \{2, 4, 8, 16, 32, 64, 128, 256, 512, 1024\} |
| LOF and LoOP | $k$: \{5, 10, 20, 30, 50, 100, 150, 200, 500, 1000\} |

*Fig. 5.* Best AUCs of different anomaly detectors on a synthetic dataset of size $n = 2499$. The best parameters here are: $k = 5$ for LOF and LoOP; $\psi = 1024$ for iForest and RMF; and $\mathcal{L} = 0.01n$ for NCAD.

### 5.1 Anomaly Detection on a Synthetic Dataset

We use a synthetic dataset to showcase the detecting power of NCAD. The top left plot in *Fig. 5* shows the data distribution with ground truth labels. The best AUCs of the five anomaly detectors and the distributions of their anomaly scores are shown in the corresponding plots in *Fig. 5*. Here NCAD has the highest AUC with almost perfect ranking of anomaly scores. LOF and LoOP have lower AUCs because the two anomalies have quite different scores due to the different curvatures. RMF has the same issue leading to low AUC, though it has substantially better AUC than iForest.
| Dataset   | $N$  | $d$ | ano%  | AUC   |   |   |   |
|----------|-----|-----|-------|-------|---|---|---|
|          |     |     |       | NCAD | ifForest | LOF | RMF | LoOP |
| breastw  | 683 | 9   | 35    | 0.994| 0.993    | 0.955| 0.947| 0.982|
| covertype| 286048 | 10  | 0.96  | 0.955| 0.912    | 0.944| 0.953| 0.790|
| diabetes | 768 | 8   | 34.9  | 0.719| 0.681    | 0.723| 0.683| 0.682|
| htru2    | 17898 | 8   | 9.2   | 0.924| 0.931    | 0.826| 0.942| 0.669|
| islet    | 351 | 32  | 35.9  | 0.896| 0.849    | 0.894| 0.887| 0.901|
| isolet   | 7797 | 617 | 3.85  | 0.871| 0.801    | 0.801| 0.758| 0.662|
| mulcross | 262144 | 4   | 10    | 1.000| 0.988    | 0.610| 0.999| 0.585|
| satellite| 6435 | 36  | 31.6  | 0.734| 0.710    | 0.793| 0.715| 0.710|
| shuttle  | 49097 | 9   | 7.15  | 0.991| 0.997    | 0.592| 0.923| 0.573|
| smtp     | 95156 | 3   | 0.03  | 0.939| 0.917    | 0.954| 0.921| 0.934|
| wdbc     | 569 | 30  | 37.3  | 0.870| 0.817    | 0.863| 0.841| 0.765|
| wilt     | 4339 | 5   | 1.7   | 0.891| 0.632    | 0.863| 0.786| 0.872|
| win/draw/loss wrt iForest | 10/0/2 | 7/1/4 | 9/0/3 | 4/1/7 |
| win/draw/loss wrt LOF      | 9/0/3 | 4/1/7 | 4/0/8 | 3/0/9 |
| win/draw/loss wrt RMF      | 11/0/1 | 3/0/9 | 8/0/4 | 4/0/8 |
| win/draw/loss wrt LoOP     | 11/0/1 | 7/1/4 | 9/0/3 | 8/0/4 |
| Average rank                | 1.58  | 3.50 | 2.67  | 3.17  | 3.92 |

### 5.2 Anomaly Detection on Benchmark Datasets

The results presented in Table 2 show that NCAD is the best performer of the five methods in terms of average rank. NCAD outperforms all other methods with a large margin: it has at least 9 wins out of 12 datasets compared to any other methods. The Nemenyi test [8] result in Fig. 6 shows that NCAD is better than iForest and LoOP at 0.05 significance level.

Note that LOF and LoOP have substantially lower AUCs than other methods on some datasets, e.g., htru2, mulcross and shuttle, shown in Table 2. This is probably due to the sensitivity of the $k$ parameter which needs to be more carefully tuned for these datasets. Searching a finer grid and a much larger range of $k$ values may improve their AUCs. However, this comes with a large expense in runtime.

### 5.3 Scale-Up Test

A scalability test is shown in Fig. 7. The test employs subsets of increasing data sizes drawn from the covertype dataset. Parameters of all methods are fixed. LOF has a quadratic time complexity while all the tree-based methods demonstrate linear trends. LoOP is not included in this test because we use the PyNomaly [7]
Python implementation while other methods are implemented in Matlab. LoOP is expected to have similar run time as LOF.

Our results are consistent with comparisons of existing anomaly detectors [2, 10] which advocate iForest in general because it is less sensitive to parameter settings and runs significantly faster than nearest-neighbour-based methods.

![Nemenyi test](image1.png)  ![Runtime](image2.png)

**Fig. 6.** Nemenyi test ($\alpha = 0.05$) of the 5 methods.

**Fig. 7.** Runtime of the four methods while data size increases on covertype. Parameter settings: $\mathcal{L} = 0.05n$ for NCAD, $k = 5$ for LOF, $\psi = 256$ for iForest and RMF.

### 6 Conclusions

Our analysis of ratio-based relative scores reveals a key shortcoming: when anomalies are located in different regions where the curvatures of the density surface are greatly different, their scores and hence the rankings are heavily impacted. This insight inspires us to use a new score based on NC, which is effective in detecting local minima of a function derived from density or other base measures.

The proposed NC anomaly score and its anomaly detector NCAD are effective in detecting both local and global anomalies, and robust against varying curvatures of local density surface. Our empirical assessment verifies its superior anomaly detection performance over iForest, LOF, RMF and LoOP on benchmark datasets. In addition, NCAD has similar time complexity as RMF and iForest, and it runs significantly faster than LOF and LoOP.

### References

1. Aggarwal, C.C.: Outlier Analysis, 2nd edn. Springer, Heidelberg (2017). [https://doi.org/10.1007/978-3-319-47578-3](https://doi.org/10.1007/978-3-319-47578-3)
2. Aggarwal, C.C., Sathe, S.: Outlier Ensembles: An Introduction. Springer, Heidelberg (2017). [https://doi.org/10.1007/978-3-319-54765-7](https://doi.org/10.1007/978-3-319-54765-7)
3. Aryal, S., Ting, K.M., Wells, J.R., Washio, T.: Improving iForest with relative mass. In: Tseng, V.S., Ho, T.B., Zhou, Z.-H., Chen, A.L.P., Kao, H.-Y. (eds.) PAKDD 2014. LNCS (LNAI), vol. 8444, pp. 510–521. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-06605-9_42

4. Bay, S.D., Schwabacher, M.: Mining distance-based outliers in near linear time with randomization and a simple pruning rule. In: KDD (2003)

5. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: LOF: identifying density-based local outliers. In: Proceedings of the ACM SIGMOD International Conference on Management of Data (2000)

6. Chen, B., Ting, K.M.: Neighbourhood contrast: a better means to detect clusters than density. In: Phung, D., Tseng, V.S., Webb, G.I., Ho, B., Ganji, M., Rashidi, L. (eds.) PAKDD 2018. LNCS (LNAI), vol. 10939, pp. 401–412. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93040-4_32

7. Constantinou, V.: PyNomaly: anomaly detection using local outlier probabilities (LoOP). J. Open Source Softw. 3(30), 845 (2018). https://doi.org/10.21105/joss.00845

8. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7(Jan), 1–30 (2006)

9. Dua, D., Graff, C.: UCI machine learning repository (2017). http://archive.ics.uci.edu/ml

10. Emmott, A., Das, S., Dietterich, T., Fern, A., Wong, W.K.: A meta-analysis of the anomaly detection problem. arXiv:1503.01158 (2016)

11. Erfani, S., Baktashmotlagh, M., Rajasegarar, S., Karunasekera, S., Leckie, C.: R1SVM: a randomised nonlinear approach to large-scale anomaly detection. In: AAAI (2015)

12. Johnson, B.A., Tateishi, R., Hoan, N.T.: A hybrid pansharpening approach and multiscale object-based image analysis for mapping diseased pine and oak trees. Int. J. Remote Sens. 34(20), 6969–6982 (2013)

13. Kriegel, H.P., Kröger, P., Schubert, E., Zimek, A.: Loop: local outlier probabilities. In: Proceedings of the ACM Conference on Information and Knowledge Management, pp. 1649–1652 (2009)

14. Lazarevic, A., Kumar, V.: Feature bagging for outlier detection. In: KDD (2005)

15. Li, K., Li, S., Ding, Z., Zhang, W., Fu, Y.: Latent discriminant subspace representations for multi-view outlier detection. In: AAAI (2018)

16. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: ICDM (2008)

17. Liu, F.T., Ting, K.M., Zhou, Z.-H.: On detecting clustered anomalies using SCiForest. In: Balcázar, J.L., Bonchi, F., Gionis, A., Sebag, M. (eds.) ECML PKDD 2010. LNCS (LNAI), vol. 6322, pp. 274–290. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15883-4_18

18. Liu, W., Hua, G., Smith, J.R.: Unsupervised one-class learning for automatic outlier removal. In: CVPR (2014)

19. Lyon, R.J., Stappers, B.W., Cooper, S., Brooke, J.M., Knowles, J.D.: Fifty years of pulsar candidate selection: from simple filters to a new principled real-time classification approach. Mon. Not. R. Astron. Soc. 459(1), 1104–1123 (2016)

20. Pang, G., Cao, L., Chen, L., Lian, D., Liu, H.: Sparse modeling-based sequential ensemble learning for effective outlier detection in high-dimensional numeric data. In: AAAI (2018)

21. Pang, G., Cao, L., Chen, L., Liu, H.: Learning homophily couplings from non-iid data for joint feature selection and noise-resilient outlier detection. In: IJCAI (2017)
22. Ramaswamy, S., Rastogi, R., Shim, K.: Efficient algorithms for mining outliers from large data sets. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 427–438 (2000)

23. Rocke, D.M., Woodruff, D.L.: Identification of outliers in multivariate data. J. Am. Stat. Assoc. 91(435), 1047–1061 (1996)

24. Xu, H., Wang, Y., Wu, Z., Wang, Y.: Embedding-based complex feature value coupling learning for detecting outliers in non-IID categorical data. In: AAAI (2019)

25. Yamanishi, K., Takeuchi, J.I., Williams, G., Milne, P.: On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. Data Min. Knowl. Disc. 8(3), 275–300 (2004)

26. Zhou, Y., Zou, H., Arghandeh, R., Gu, W., Spanos, C.J.: Non-parametric outliers detection in multiple time series a case study: power grid data analysis. In: AAAI (2018)

27. Zimek, A., Gaudet, M., Campello, R.J., Sander, J.: Subsampling for efficient and effective unsupervised outlier detection ensembles. In: KDD (2013)