

Depth Deviation Score: A New Outlier Detection Framework

Channamma Patil., Ishwar Baidari*

Department of Computer Science, Karnatak University, Dharwad, Karnataka – 580003, India

*Corresponding Author

DOI: <https://doi.org/10.51584/IJRIAS.2026.11080116>

Received: 24 August 2026; Accepted: 29 August 2026; Published: 15 September 2026

ABSTRACT

An outlier is a data point that differs significantly from the rest of the data. Outliers are also referred to as abnormalities, discordant observations, deviants, or anomalies in the data mining and statistics literature. Detecting outliers is very useful for improving data quality, as outliers can influence the results of data mining techniques such as clustering, classification, and association. This paper proposes a simple method for detecting outliers in a dataset based on data depth. Extensive evaluation experiments conducted on real-world datasets, comparing the proposed method with popular outlier detection algorithms, show that its performance is competitive and promising.

Keywords: Data depth; Local outlier factor; Outlier detection

INTRODUCTION

An outlier is a data point that differs significantly from the rest of the data. Hawkins [1] formally defined the concept of an outlier as follows: *An outlier is an observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism.* Outliers are also referred to as abnormalities, discordants, deviants, or anomalies in the data mining and statistics literature. In most applications, data are generated by one or more processes, which may reflect system activity or observations collected about entities. When a generating process behaves in an unusual way, it results in the creation of outliers. Therefore, an outlier often contains useful information about abnormal characteristics of the systems and entities that affect the data generation process. Recognizing such unusual characteristics provides valuable application-specific insights. Examples include intrusion detection systems, credit card fraud detection, medical diagnosis, law enforcement, and earth science. In all these applications, the data have a normal model, and anomalies are recognized as deviations from this normal model. Identifying observations that are inconsistent with normal data is of great importance. The research problem underlying these applications is commonly referred to as outlier detection, novelty detection, anomaly detection, or fault detection [2].

A large body of outlier detection methods has been developed. Technically, the procedure for detecting outliers consists of two main stages: (1) outlier ranking and (2) threshold determination. The former produces a ranked list of observations, each with a score based on given metrics. Observations with high scores are placed at the top of the list when a larger score indicates a greater degree of deviation or anomalousness. The latter determines outliers based on the ranked list. From this perspective, outlier ranking plays a central role in detection. The terms outlier ranking and outlier detection are often used interchangeably in the literature. Outlier detection algorithms can be roughly categorized into several groups, including statistics-based, distance-based, density-based, and clustering-based methods [3]. Among these, distance-based and density-based detection methods have received special attention and have been extensively studied because their concepts are intuitive and easy to implement. In this evaluation study, we focus on distance-based and density-based outlier detection strategies that require a user-supplied parameter k to fix the size of the neighborhood set, referred to as the k NN set or k NNs. Distance or density metrics often fail to capture neighborhood information because, as dimensionality increases, all distances between observations tend to become indiscernible. Moreover, finding the

kNNs in a dataset of size N by linear scan requires $O(N^2)$ distance computations, so the cost of finding neighbors usually dominates the algorithm's runtime.

Distance-based detection methods consider only global information, and their performance depends on the size of the neighborhood, while density-based methods are sensitive to the parameters that define the neighborhood [3, 4].

In this paper, we attempt to address the above problems by developing a novel yet effective learning method for outlier detection based on data depth. The proposed method is motivated by the simple idea that anomalous observations have low depth values, while the central point has the maximum depth value. The deviation of observations from the central observation is the key factor used to compute a score for each observation; based on these scores, observations are classified as outliers or not.

Our method starts by computing the depth for each observation. From these depth values, the maximum depth is selected as the depth median (the central value). Then, the deviation of each observation from this central point is calculated and used as its scoring value. Subsequently, all observations are ranked in descending order according to their scores. Finally, observations with high scores are selected as outliers. The main contributions of this paper are highlighted as follows:

1. We propose a novel and effective outlier detection method capable of handling high-dimensional data.
2. Unlike the methods discussed in this study, our method does not require any input parameter k .
3. Our method adopts a new anomalous score, called the depth deviation score, to capture the degree of deviation of an observation from the central point.

The rest of this paper is organized as follows. Section 2 briefly reviews previous research on outlier detection. Section 3 provides the basic concepts used and describes the proposed outlier detection method. Section 4 presents the experimental results of our method compared with benchmark algorithms on real datasets, followed by the conclusions in Section 5.

Background and Related work

Over the years, a great number of outlier detection algorithms have been proposed in the research community. Outlier detection methods can be roughly classified into supervised, semi-supervised, and unsupervised scenarios, depending on the availability of data labels. Supervised methods require data objects tagged as either normal or abnormal, while in the semi-supervised context, only normal objects are labeled. In unsupervised techniques, label information is unavailable. Since obtaining label information in real-world applications is very expensive, unsupervised techniques are more widely applicable than supervised ones. Based on the techniques adopted, outlier detection algorithms can be roughly classified as statistics-based, clustering-based, distance-based, density-based, and so on [3]. The basic idea of distance-based detection algorithms is that an observation is likely to be regarded as an outlier if it is far from its nearest neighbors. Essentially, these methods first find the nearest neighbors for each observation and then estimate the distance of the observation to its neighbors. Subsequently, the distances for all observations are ranked, and observations with larger distances are considered outliers. A typical example is the local distance-based outlier factor (LDOF) [5]. Since no assumption about the data distribution is required, distance-based detection algorithms have been extensively studied. The performance of distance-based detection methods greatly depends on the definition of distance and the efficiency of nearest neighbor search. Several attempts have been made in these directions. In [6], the distance from an observation to its kNNs (i.e., the maximum distance) was used, while in [7], the sum of distances from an observation to its k neighbors was calculated. Ha et al. [8] estimated the distance to the center of gravity of the neighbors, which represents a geometric property of an observation. Koufakou and Georgiopoulos [9] discussed distance definitions for mixed-type features. Liu and Deng [10] extended the classical LOF to uncertain data. Wang et al. [11] exploited a minimum spanning tree to improve the efficiency of neighbor search in kNN. Other work has focused on using variants of kNN to improve performance. Recently, Huang et al. [12] adopted the concept of natural neighbor to obtain neighborhood information, while Radovanovic et al. [13] employed

reverse nearest neighbors instead of nearest neighbors to identify outliers. A major challenge for distance-based detection algorithms is high dimensionality [4], where data are often sparsely distributed and points become similar to each other, resulting in small differences in actual distances between many pairs of observations [14]. In other words, as dimensionality increases, the distance from an observation to its nearest neighbor becomes close to the distance to its farthest neighbor. Hence, the discriminative power of distance-based techniques diminishes, especially when data come from a mixture of distributions with varying cluster densities. In the literature, subspace learning [4], random sampling [15], and feature selection [16, 17] are three frequently used strategies to alleviate this problem. Additionally, because distances are calculated based on global information rather than local information, another problem with distance-based methods is that they can only identify global outliers and fail to detect local ones.

The underlying principle of density-based detection approaches is that an outlier lies in a low-density neighborhood, while a normal observation has a dense neighborhood. Specifically, these methods first estimate the neighborhood density for each observation and then compare it with the densities of the observation's neighbors. If the density differs considerably from that of its neighbors, the observation can be declared an outlier. Due to their effectiveness and simplicity, density-based detection approaches have been widely used in practice. LOF [18] is a representative example. It uses local density, estimated from the reachability distance of kNNs, to measure the degree of outlierness. After LOF was introduced, several variants were developed, including the connectivity-based outlier factor [19], the influenced outlierness measure [20], the local correlation integral [21], and local outlier probabilities [22]. Although these methods are very popular, like distance-based methods, they rely on distance computations and face the same challenges posed by high-dimensional data, where outlier scores become close to each other. Density-based methods are sensitive to the parameters used to determine the neighborhood size and show poor performance when observations have varying densities. The three popular outlier detection algorithms used in the comparative study are explained below.

Local Outlier Factor (LOF) was the first local outlier method. It compares the local reachability density (lrd) of the kNN set of the test point with those of the neighborhoods of each member of the kNN set; a high value indicates outlierness. Local reachability density is defined as the inverse of the average reachability distance from the neighbors. LOF is a popular density-based algorithm. Due to its high time complexity, LOF is not suitable for large-scale high-dimensional datasets.

Local Outlier Probabilities (LOOP) uses a more robust local density estimate based on the quadratic mean distance. Furthermore, LOOP normalizes the outlier detection scores.

Local Distance-based Outlier Factor (LDOF) defines outlierness as the ratio of the average distance from the test point to its kNNs to the average pairwise distance within the kNN set. LDOF favors outliers that are far from a compact kNN set. Like COF, LDOF requires computing all pairwise distances ($O(k^2)$).

Proposed method

Data Depth

Data depth is a statistical function that measures the deepness or centrality of a point in a data cloud. A larger depth value indicates that a point is deeper within the data cloud, while a lower value indicates outlyingness. Data depth can also provide information about the spread, shape, and symmetry of a dataset through depth regions [23]. Statistical depth functions provide an ordering of all points from the center outward in a multivariate dataset. Tukey [24] first introduced half space depth. Oja [25] defined Oja depth, and several other depth functions have been proposed in the literature, such as convex-hull peeling depth [26], simplicial depth [27], regression depth [28], L1 depth [29], and Mahalanobis depth [30]. The four desirable properties introduced by Liu [27] for depth functions are affine invariance, maximality at the center, monotonicity relative to the deepest point, and vanishing at infinity. More details about depth functions and their properties can be found in Zuo and Serfling [30]. Mahalanobis depth is one of the well-known depth functions that we use in our proposed method. It satisfies all four properties mentioned above. It is fast and easy to compute, and we consider this depth function to be competitive.

Mahalanobis Depth

The Mahalanobis depth (MD) function was proposed by Liu and Singh [31] and is based on the well-known Mahalanobis distance defined by Mahalanobis [32]. The MD of a point x with respect to a dataset X is defined as:

$$MD(x | X) = [1 + (x - \bar{x})^T \text{Cov}(X)^{-1} (x - \bar{x})]^{-1} \quad (1)$$

where $\bar{x}$ and $\text{Cov}(X)$ are the mean and covariance matrix of X , respectively. The point with the maximum depth value is the deepest point in the dataset; a larger value indicates a deeper point, and a smaller value indicates outlyingness, as shown in Figure 1.

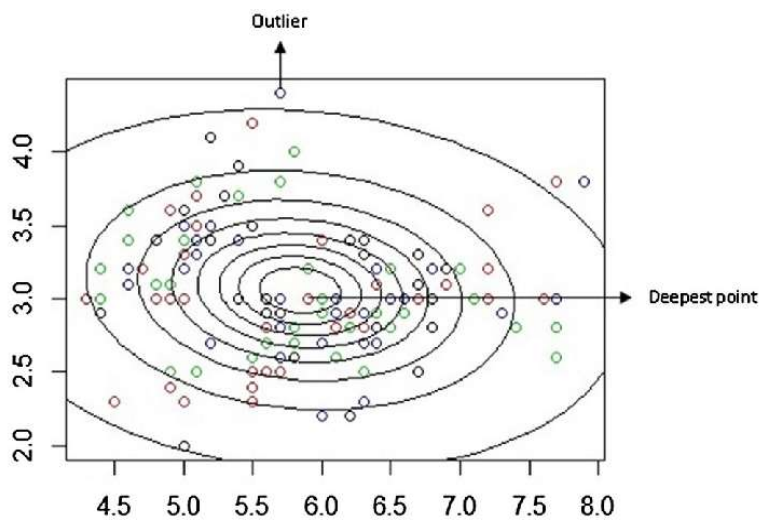


Fig. 1: Mahalanobis Depth Contours over Iris Dataset

However, this depth function is not robust because it is based on the sample mean and covariance matrix. Therefore, to calculate the depth of each point, we use robust alternatives for the mean vector and covariance matrix, such as Rousseeuw's minimum covariance determinant (MCD) or the minimum volume ellipsoid (MVE) estimates, which are available in the literature [33, 34]. The robust MD of a point x with respect to a dataset X is defined as:

$$MD(x; X) = [1 + (x - X_i)^T C^{-1} (x - X_i)]^{-1} \quad (2)$$

Here, X_i and C are the MCD-based or MVE-based estimators of the mean vector and the covariance matrix, respectively.

In this section, we propose a new method for detecting outliers in a dataset based on data depth. In the following Section 3.2, we explain the definitions used in the method.

Definitions Used in the Proposed Method

Let $X = \{x_1, x_2, \dots, x_n\}$ be a dataset with n instances. A data depth function assigns a value between 0 and 1 to each data point in the dataset, specifying the centrality or deepness of that point in the dataset. The depth of each point x_i ($i = 1, 2, \dots, n$) in dataset X is calculated using Eq. (2) and is denoted by $D(x_i|X)$.

Definition 1.

Depth median (μ): We define the maximum depth value in X as the depth median, denoted by μ , formulated as follows:

$$\mu = \max_{1 \leq i \leq n} D(x_i | X) \quad (3)$$

where n is the number of points in X .

Definition 2.

Depth deviation ($d(x_i)$): We define the difference between the depth of each object and the depth of the center point as the depth deviation, formulated as follows:

$$d(x_i) = \mu - D(x_i | X) \quad (4)$$

Algorithm

1. The algorithm calculates the depth $D(x_i|X)$ of each point x_i in dataset X using Mahalanobis depth.
2. From the computed depths, the algorithm finds the maximum depth (μ), or depth median, of dataset X , which is the central point.
3. The algorithm calculates the deviation of each observation from the central point. This deviation value is considered the ranking score of the observation.
4. The algorithm sorts the observations in descending order based on the deviation values.
5. From the sorted observations, the top s observations are the desired outliers

Algorithm 1 Outlier Detection

- **Input:** A dataset $X = \{x_1, x_2, \dots, x_n\}$, and the number of outlier candidates s
 - **Output:** The s desired outliers
1. **For each** observation $x_i \in X$ **do**
 2. Calculate the depth $D(x_i | X)$ of x_i using Eq. (2)
 3. Find the depth median μ of X using Eq. (3)
 4. Find the depth deviation $\delta(x_i)$ using Eq. (4)
 - **End For**
 5. Sort depth deviation values in descending order
 - 9: **Return** the top s observations as the desired outliers

Experiments

To verify the performance of the proposed algorithm, we conducted experiments on eleven real-world datasets and compared the results with three popular outlier detection algorithms. In this section, we discuss the details of the datasets, the evaluation metric used, and the results obtained.

Datasets The experiments were conducted on eleven real-world datasets: Wine, Lymphography, Stampout, Ionosphere, WBC, Arrhythmia, Pima, Banknote, Cardio, Annthyroid, and Shuttle. The details of these datasets

are shown in Table 1. These datasets are frequently used in the literature to test the performance of detection methods. All datasets were selected from the Outlier Detection Datasets website [35]. The original datasets are from the UCI Machine Learning Repository [36] and were originally used for classification. To use these datasets for outlier detection, the minority class in each dataset was considered as outliers, and the remaining data as normal. After this reformulation, they were used to evaluate the performance of outlier detection methods. For example, in the Annthyroid dataset, three classes are present: normal (not hypothyroid), hyperfunction, and subnormal functioning. For outlier detection, the hyperfunction and subnormal classes were treated as the outlier class, and the other class as inliers. In Ionosphere, the "bad" class was considered outliers and the "good" class as inliers. The Lymphography dataset is a multi-class dataset with four classes, but two of them are quite small; therefore, those two small classes were merged and considered as outliers compared to the other two large classes. The Cardiotocography (Cardio) dataset has three classes: normal, suspect, and pathologic. For outlier detection, the normal class formed the inliers, while the pathologic (outlier) class was downsampled to 176 points. The suspect class was discarded. The Pima dataset was used as is from the UCI repository. In the Wisconsin Breast Cancer (Diagnostics) dataset (WBC), there are two classes: benign and malignant. The malignant class was downsampled to 21 points, which were considered outliers, while points in the benign class were considered inliers. In the Shuttle dataset, the smallest five classes (2, 3, 5, 6, 7) were combined to form the outlier class, while class 1 formed the inlier class. Data for class 4 were discarded. In the Arrhythmia dataset, the smallest classes (3, 4, 5, 7, 8, 9, 14, 15) were combined to form the outlier class, and the remaining classes were combined to form the inlier class. In the Wine dataset, classes 2 and 3 were used as inliers, and class 1 was downsampled to 10 instances to be used as outliers.

Dataset	No. of Instances	No. of Attributes	No. of Outliers
Wine	129	13	10
Lymphography	148	18	6
Stampout	340	9	31
Ionosphere	351	33	126
WBC	378	30	21
Arrhythmia	452	274	66
Pima	768	8	268
Banknote	1372	4	610
Cardio	1831	21	176
Annthyroid	7200	6	534
Shuttle	49097	9	3511

Table 2: Precision (%) of the outlier detection algorithms on the experimental datasets

Dataset	Top s	LOF	LDOF	LOOP	Depth
Wine	20	15.0	10.0	10.0	5.0
	50	16.0	4.0	4.0	14.0



	100	9.0	5.0	10.0	10.0
Lymphography	20	30.0	11.0	25.0	20.0
	50	12.0	6.0	12.0	12.0
	100	6.0	6.0	6.0	6.0
Ionosphere	20	95.0	90.0	95.0	100.0
	50	96.0	82.0	90.0	100.0
	100	86.0	70	79	93.0
WBC	20	20.0	30.0	20.0	45.0
	50	16.0	20.0	18.0	32.0
	100	9.0	5.0	10.0	10.0
Arrhythmia	20	30.0	20.0	30.0	75.0
	50	24.0	20.0	09.0	50.0
	100	19.0	16.0	15.0	40.0
Pima	20	40.0	30.0	40.0	60.0
	50	32.0	40.0	34.0	58.0
	100	30.0	37.0	29.0	49.0
Banknote	20	05.0	10.0	00.0	90.0
	50	20.0	12.0	14.0	74.0
	100	24.0	23.0	18.0	67.0
Cardio	20	20.0	05.0	15.0	45.0
	50	26.0	16.0	16.0	70.0
	100	23.0	17.0	15.0	72.0
Anthyroid	20	00.0	00.0	10.0	55.0
	50	00.0	04.0	08.0	40.0
	100	22.0	10.0	05.0	34.0
Shuttle	20	45.0	00.0	30.0	55.0
	50	30.0	14.0	28.0	64.0
	100	20.0	14.0	18.0	53.0

Evaluation Metric

To measure the performance of all outlier detection algorithms on the experimental datasets, we selected precision (P_r) as the evaluation metric. This metric is frequently used in the literature to measure the performance of learning algorithms. It refers to the ratio of the number of true outliers detected by an outlier detection algorithm to the total number of outlier candidates, i.e.:

$$P_r = k / s \quad (5)$$

where k is the number of true outliers found within s outlier candidates. This criterion is also called precision@ k in the literature, because s is fixed during the evaluation experiments [2].

RESULTS

In the experiments, the proposed method was compared with three popular outlier detection algorithms: LOF [18], LDOF [5], and LOOP [22]. The experiments were carried out on an Intel i3 core processor with 8 GB of RAM running Windows 8.0. In the comparative experiments, we first evaluated the performance of the outlier detection algorithms using the precision criterion. Specifically, we ran each outlier detection algorithm on each dataset and ranked the observations according to the corresponding scores. Then we took the top s ($s = 20, 50$, and 100 , respectively) candidate outliers. Among these, the number of true outliers was counted to estimate precision. Table 2 reports the precision (%) achieved by the compared algorithms when the top s ($s = 20, 50$, and 100) outlier candidates were selected.

In Table 2, bold values indicate the best performance among the compared algorithms on each dataset (for the same row). For example, our method achieved the best performance on all datasets except the Wine dataset. From the experimental results, one can observe that compared to the popular outlier detection algorithms, our method shows predominant performance in identifying outliers.

Table 3 shows the elapsed time of the outlier detection algorithms on the experimental datasets. In the experiments, the k value was set to 5 for LOF, LDOF, and LOOP, while the proposed method does not require any input parameter. It can be observed from Table 3 that the LDOF algorithm has the highest elapsed time on all datasets. LOF and LOOP also have high elapsed times compared to the proposed method. The proposed method has the lowest elapsed time on all datasets. The run times in Table 3 show that compared to the popular outlier detection algorithms, the efficiency of the proposed method is promising.

Table 3: Time cost (in seconds) of the outlier detection algorithms on the experimental datasets

Dataset	LOF	LDOF	LOOP	Depth
Wine	0.13	0.11	0.14	0.08
Lymphography	0.12	0.13	0.12	0.07
Ionsphere	0.22	0.36	0.22	0.07
WBC	0.25	0.36	0.23	0.07
Arrhythmia	0.44	1.1	0.37	0.26
Pima	0.42	1.95	0.28	0.07
Banknote	0.65	4.2	0.52	0.07

Cardio	1.1	7.5	0.85	0.07
Annthroid	2.93	120	1.95	0.1
Shuttle	20.75	30	14	1.14

CONCLUSION

In this paper, we developed an efficient and effective learning method to identify outliers from normal observations. The main idea of the proposed method is to exploit the deviation of an observation from the central point to determine whether it is an outlier. To find the deviation of an observation, we first calculated the depth of each point in the dataset. The point with the maximum depth is the deepest or central point in the dataset. We then computed the deviation of each point by subtracting its depth from the depth of the central point. Observations with high deviation are promising candidates for being outliers with high probability. Moreover, unlike existing distance-based and density-based detection methods, the proposed method does not require an input parameter k . To demonstrate the effectiveness of our method, we conducted comprehensive experiments comparing it with three popular outlier detection algorithms on several public real-world datasets. The numerical results show that the proposed method is good at ranking the best candidates for outliers, and its performance is promising.

REFERENCES

1. Hawkins DM, Identification of Outliers, Springer, (1980).
2. Allen LV & Tilbury DM, "Anomaly detection using model generation for event-based systems without a preexisting formal model", IEEE Trans. Syst., Man, Cybern. A, Syst., Humans, Vol.42, No.3, (2011), pp.654-668.
3. Chandola V, Banerjee A & Kumar V, "Anomaly detection: A survey", ACM Comput. Survey, Vol.41, No.3, (2009), pp.1-58.
4. Aggarwal CC, Data Mining, New York, NY, USA: Springer, (2013)
5. Zhang K, Hutter M & Jin H, "A new local distance-based outlier detection approach for scattered real-world data", Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, (2009), pp.813-822.
6. Ramaswamy S, Rastogi R & Shim K, "Efficient algorithms for mining outliers from large data sets", Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, (2000), pp.427-438.
7. Angiulli F & Pizzuti C, "Fast outlier detection in high dimensional spaces", Proceedings of the European Conference on Principles of Data Mining and Knowledge Discovery, (2002), pp.15-27.
8. Ha J, Seok S & Lee JS, "Robust outlier detection using the instability factor", Knowledge-Based Systems, Vol.63, (2014), pp.15-23.
9. Koufakou A & Georgiopoulos M, "A fast outlier detection strategy for distributed high-dimensional data sets with mixed attributes", Data Mining and Knowledge Discovery, Vol.20, No.2, (2010), pp.259-289.
10. Liu J & Deng HF, "Outlier detection on uncertain data based on local information", Knowledge-Based Systems, Vol.51, (2013), pp.60-71.
11. Wang X, Wang XL, Ma Y & Wilkes DM, "A fast MST-inspired kNN-based outlier detection method", Information Systems, Vol.48, (2015), pp.89-112.
12. Huang J, Zhu Q, Yang L & Feng J, "A non-parameter outlier detection algorithm based on natural neighbor", Knowledge-Based Systems, Vol.92, (2016), pp.71-77.
13. Radovanović M, Nanopoulos A & Ivanović M, "Reverse nearest neighbors in unsupervised distance-based outlier detection", IEEE Transactions on Knowledge and Data Engineering, Vol.27, No.5, (2014), pp.1369-1382.
14. Zimek A, Schubert E & Kriegel HP, "A survey on unsupervised outlier detection in high-dimensional numerical data", Statistical Analysis and Data Mining: The ASA Data Science Journal, Vol.5, No.5, (2012), pp.363-387.

15. Ha J, Seok S & Lee JS, "A precise ranking method for outlier detection", *Information Sciences*, Vol.324, (2015), pp.88-107.
16. Xiao C & Chaovalitwongse WA, "Optimization models for feature selection of decomposed nearest neighbor", *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol.46, No.2, (2015), pp.177-184.
17. Wu Q, Wang Z, Deng F, Chi Z & Feng DD, "Realistic human action recognition with multimodal feature selection and fusion", *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol.43, No.4, (2013), pp.875-885.
18. Breunig MM, Kriegel HP, Ng RT & Sander J, "LOF: identifying density-based local outliers", *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, (2000), pp.93-104.
19. Tang J, Chen Z, Fu AW & Cheung DW, "Enhancing effectiveness of outlier detections for low density patterns", *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, (2002), pp.535-548.
20. Jin W, Tung AKH, Han J & Wang W, "Ranking outliers using symmetric neighborhood relationship", *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, (2006), pp.577-593.
21. Papadimitriou S, Kitagawa H, Gibbons PB & Faloutsos C, "LocI: Fast outlier detection using the local correlation integral", *Proceedings of the 19th International Conference on Data Engineering*, (2003), pp.315-326.
22. Kriegel HP, Kröger P, Schubert E & Zimek A, "LoOP: local outlier probabilities", *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, (2009), pp.1649-1652.
23. Serfling R, "Nonparametric multivariate descriptive measures based on spatial quantiles", *Journal of Statistical Planning and Inference*, Vol.123, No.2, (2004), pp.259-278.
24. Tukey JW, "Mathematics and the picturing of data", *Proceedings of the International Congress of Mathematicians*, Vol.2, (1975), pp.523-531.
25. Oja H, "Descriptive statistics for multivariate distributions", *Statistics & Probability Letters*, Vol.1, No.6, (1983), pp.327-332.
26. Eddy WF, "Convex hull peeling", *Proceedings of the COMPSTAT 1982 5th Symposium*, (1982), pp.42-47.
27. Liu RY, "On a notion of data depth based on random simplices", *The Annals of Statistics*, (1990), pp.405-414.
28. Rousseeuw PJ & Hubert M, "Regression depth", *Journal of the American Statistical Association*, Vol.94, No.446, (1999), pp.388-402.
29. Vardi Y & Zhang CH, "The multivariate L₁-median and associated data depth", *Proceedings of the National Academy of Sciences*, Vol.97, No.4, (2000), pp.1423-1426.
30. Zuo Y & Serfling R, "General notions of statistical depth function", *Annals of Statistics*, (2000), pp.461-482.
31. Liu RY & Singh K, "A quality index based on data depth and multivariate rank tests", *Journal of the American Statistical Association*, Vol.88, No.421, (1993), pp.252-260.
32. Mahalanobis PC, "On the generalized distance in statistics", *Proceedings of the National Institute of Sciences of India*, Vol.12, (1936), pp.49-55.
33. Rousseeuw PJ, "Multivariate estimation with high breakdown point", *Mathematical Statistics and Applications*, Vol.8, (1985), pp.283-297.
34. Rousseeuw PJ & Leroy AM, *Robust Regression and Outlier Detection*, John Wiley & Sons, (2003).
35. Rayana S, "ODDS Library", Stony Brook University, (2016), available online: <http://odds.cs.stonybrook.edu>, last visit: 21.07.2026.
36. Dua D & Graff C, "UCI Machine Learning Repository", University of California, (2017), available online: <http://archive.ics.uci.edu/ml>, last visit: 21.07.2026.