Indian Journals of Science and Technology, Vol 9(19), DOI: 10.17485/ijst/2016/v9i19/93884, May 2016
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

An Efficient Unsupervised Clustered Adaptive Antihub Technique for Outlier Detection in High Dimensional Data

R. Lakshmi Devi[1*] and R. Amalraj[2]

[1]Mother Teresa Women's University, Kodaikanal, India; surlakp@yahoo.co.in
[2]Department of Computer Science, Sri Vasavi College, Erode, India; r_amalraj05@yahoo.co.in

Abstract

Objective: The objective of this paper is to find the inconsistent objects in data which has high dimension through reduced computation time and increased accuracy. **Methods:** Hubness specifically Antihubs (points that rarely occur in k nearest neighbor lists) is the newly recognized concept for handling data which has high dimension. The advanced version of Antihub is Antihub2 which is for reconsidering the outlier score of a point obtained by the Antihub method. However, regarding computation time, Antihub2 runs slower. This paper institutes an approach called AdaptiveAntihub2Clust, which is a clustered Adaptive Antihub technique for unsupervised outlier detection to reduce computation time and to improve the accuracy. **Findings:** The results of an existing Antihub2 method is compared with the proposed method called AdaptiveAntihub2Clust. The experimental results elucidate that AdaptiveAntihub2Clust outperforms well than Antihub2 and also resolved that there is not only a substantial decrease in computation time but also progress in accuracy occurred while the newly built approach is practically used for finding outliers. **Applications:** The irrelevant objects may ascend due to numerous faults. Detection of such objects identifies the mistakes and fraud before they deteriorate with terrible significances and cleanses the data for further processing.

Keywords: Adaptive Antihub, Antihub, Antihub2, Outliers, Unsupervised

1. Introduction

Finding inconsistent objects is the procedure of identifying observations in data which do not follow usual behavior[1]. Three sorts of outliers used are point, contextual and collective. Based on the accessibility of label, finding irrelevant object able to be in one of three categories such as supervised, semi supervised and unsupervised. The unsupervised category is more suitable which does not need tagging. The labeling is required for the further two techniques to produce the training set which is an expensive and time consuming[2].

Hubness is derived from the notion of k occurrences and considered as a feature of the increase in dimension related to neighbors which is nearest[3]. The recent research papers deal with low point in knn lists for handling data when the dimension is high. Amount of reverse nearest neighbors is acknowledged in[4] for those which do not involve tag along with the consideration of distance. The mark for rare object considered with this count is presented as an antihub where a user defined threshold is compared with the inconsistent object score to regulate if the object is an inconsistent or not. Proved the appearance of the hubness in [5] and presenting that it is a noteworthy property for the data when the dimension is high.

The approach for discovering outliers which are assembled on distance built upon the k nearest neighbor points is proposed in paper[6]. Density based degree can be used in [7] for distinguishing occasional objects from the consistent objects. Similarly LDOF (Local Distance-based Outlier

*Author for correspondence

Factor)[8] and LoOP (Local Outlier Probability)[9] are also used for the same purpose. Algorithms that dealt with the similar factors show a dynamic role in detecting outliers[10,11].

Clustering is a standard technique used to group related objects in groups or clusters[12]. Among the various clustering algorithms, K-means is a commonly used one and also considered as one among the top ten algorithms in data mining[13].

K-means clustering is explored in [14] for the detection of irrelevant objects in measurement of software data. In [15] clustering algorithm K-means is utilized as a tool to set consistent and inconsistent traffic in network. Similarly radius of every group is used for defining the unequal objects in [16, 17] and [18] also introduced the k-means for grouping regular and irregular objects.

Since Clustering displays a role in handling data which has high dimension especially for outlier detection, it is concentrated by way of using hubness. The hubness (Antihub2) is interested to relate the newly projected method called AdaptiveAntihub2Clust where hubness, i.e adaptive method applied in antihub is fixed in the subsequent cluster groups gotten from clustering method K-means to perceive the outliers.

The remaining paper is designed as follows: Materials and methods are stated in section 2. Projected method and its suggestions are elucidated in section 3. The results and discussions are described in section 4. Section 5 labels conclusion.

2. Materials and Methods

2.1 K-means Algorithm

Clustering is the development of grouping n object into K clusters. Let X= $\{x_i\}$ i=1; . . . ; n be the data set and X is to be collected into a set of K clusters, C = $\{c_k, k = 1 \ldots ; K\}$. This algorithm catches a partition such that the squared error between the mean of a cluster and each object in cluster is lessened. If c_k is a cluster, μ_k is the mean of cluster. The error is well-defined as

$$J(C_k) = \sum_{x_i \in ck} \|x_i - \mu_k\|^2$$

K-means is to minimize the sum of the squared error over all K clusters,

$$J(C) = \sum_{k=1}^{K} \sum_{x_i \in ck} \|x_i - \mu_k\|^2$$

To start with, initial state is selected with K clusters and partition is made by transferring each object to its nearest center of cluster. New cluster center is computed. This procedure is repeated until all objects are grouped.

2.2 Antihub

Antihub has newly developed as a main aspect with respect to nearest neighbors. Hubness is derived from the notion of k occurrences. Hubs and antihubs differ only in the quantity in kNN sets, where hubs have high quantity of points and antihubs have less quantity of points or none. In particular, hubness denotes to a growing skewness in the k occurrence distribution in high-dimensional data[19].

The algorithm Antihub is to catch irrelevant objects. For each object x in the ordered data set, find the k count of reverse nearest neighbors $N_k(x)$ for each and every object with respect to distance measured by Euclidian. Find the inconsistent object score which is $1/ (N_k(x) + 1)$ for each object. There may be an irrelevant object if the score is higher. According to the user definite threshold value, irrelevant object is found.

2.3 Antihub2

Antihub, which states the inconsistent object score of object x from data set D as a function of $N_k(x)$. The scores created by Antihub are distinct irrespective of dimensionality. Discrimination of scores characterizes a noteworthy weakness of the Antihub algorithm. Antihub, which improves outlier scores formed by the Antihub algorithm by also considering the N_k scores of the neighbors of x, in addition to $N_k(x)$ itself.

For each object it finds anni which is the summation of outlier score for each object. It finds the ct value by calculating $(1 - \alpha) \bullet a_i + \alpha \bullet ann_i$ where ai is the antihub score also calculates the cdisc. cdisc is discScore (ct, p) where $p \in (0, 1]$ outputs the amount of exclusive items among $\lceil np \rceil$ lowest members of ct, divided by $\lceil np \rceil$. By comparing disc and cdisc values, corresponding ct and cdisc values are assigned to t and disc respectively. Irrelevant object score is attained for each object and there is a chance of finding

irrelevant object if the score is higher. According to the user definite threshold value, irrelevant object is found.

3. Proposed Method

The proposed approach AdaptiveAntihub2Clust is a clustered Adaptive Antihub by which time computed is to be reduced and the accuracy is to be improved. The basic structure of the newly projected method is specified in Figure 1.

This paper institutes an approach called AdaptiveAntihub2Clust which is a clustered adaptive antihub technique[20] where cluster based adaptive technique is applied in Antihub2 algorithm which is a progressive form of Antihub which considers not only the reverse k-nearest neighbor count of x but also reverse k-nearest neighbor count scores of the neighbors of x, are taken for inconsistent object detection. In data which is high in dimension, for all objects x, each α is used and α is also designed by the step size where step ∈ (0, 1), it may lead Antihub2 to complete lengthy period.

In contrast to this, as an alternative of using all α values for each point x, the proposed approach is required not only to find the best α value but also to reduce the computation time. In the proposed system AdaptiveAntihub2Clust, there are two steps involved.

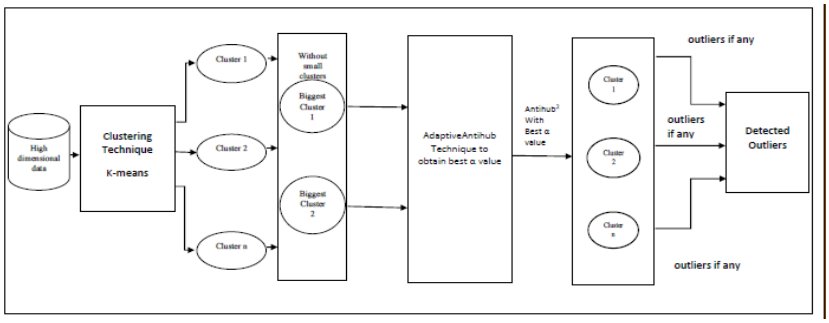

Figure 1. Basic structure of the proposed approach.

To begin with the step1, data is split into various collection of objects with the support of k means. Small clusters (minimal facts than ½ of the average amount of facts in the k clusters) are taken into account and reflected as inconsistent objects[21]. On account of this the time computed is considerably decreased. In step 2, subsequent collections are treated as an input and for the two biggest clusters, it finds the best α value by dividing the α set into four sectors and comparing the corresponding cdisc values of these sectors, it moves into the corresponding direction quickly by which it finds the best α value to reduce computation time. Once it finds the best α value that is applied along with Antihub2 to all the remaining clusters so that it further reduces computation time as well as increase the accuracy.

The basic structure of the proposed algorithm is as follows:

AdaptiveAntiHub2Clust$_{dist}$

Steps:

1. Apply kmeans algorithm to data which is high in dimension to generate clusters.
2. Regulate small clusters and consider the points that belong to these clusters as irrelevant objects and prune them out. Go to 3.
   Else
3. For the two biggest clusters
4. For each object
5. Estimate anni which is the summation of outlier score for each object.
6. disc: = 0
7. For each α ∈ (0, step, 2 • step . . . 1)
8. Set sbeg =1; send=no.of α values; divv=send/4; to split the α set into four segments
   and evaluate lmid=round (sbeg + divv); rmid=round (send – divv);
9. while lmid<= rmid && lmid>= sbeg
   {
10. Estimate α (lmid) and α (rmid) as alphalmidv and alpharmidv individually
11. For each i∈ (1, 2 . . . c_n)
12. Define the ct_i values (lct and rct) centered on alphalmidv and alpharmidv

respectively where cti: = $(1 - \alpha) \bullet a_i + \alpha \bullet ann_i$

13. Define lmcdisc and rmcdisc values according to lct and rct values
    correspondingly where cdisc: = discScore (ct, p) where $p \in (0, 1]$ outputs the
    quantity of exclusive items among ⌈np⌉ lowest members of ct, divided by ⌈np⌉
14. If lmcdisc is equal to rmcdisc
15. Allocate lct and lmcdisc to t and disc respectively
    Else if lmcdisc is less than rmcdisc
    Allocate rct and rmcdisc to t and disc respectively and
    Estimate lmid = round (lmid + divv);
    Else if lmcdisc is greater than or equal to rmcdisc
    Allocate lct, lmcdisc and lmid to t, disc and rmid respectively and
    Estimate lmid = round (lmid – divv);

}

16. For each $i \in (1, 2 \ldots c_n)$
17. s_i: = f (t_i), where f: R → R is a monotone function
18. Find the best α conforming to best accuracy from the two biggest clusters.
19. Apply Antihub2 with the best α for the left over clusters to obtain the outliers.

4. Results and Discussion

Clustered adaptive antihub technique is utilized for the statistical procedures of accuracy and elapsed time estimation. Computation time and accuracy studies efficiency of the projected method with an assistance of three real data sets. The first one is wilt which is 4339 number image sections and comprises 6 attributes in number. The second one is aloi which is a collection of randomly 2300 image objects and 64 number of attributes. Finally churn has 1667 number of objects and 21 attributes in number. Accuracy is known to be the amount of factual grades. Genuineness of a test is proceeded here with the usage of accuracy.

Table 1 depicts time consumed by the algorithms for the k value of 120. Comparatively AdaptiveAntihub2Clust has reduced 63.40 % of its computation time in an average than the existing one. AdaptiveAntihub2Clust overtakes in performance than the other.

Table 1. Computation time of algorithms when k=120

	Antihub2 (secs)	Adaptive AntiHub2Clust(secs)
ALOI	3.6426	2.0163
WILT	9.3879	2.6836
CHURN	1.8552	0.7479
AVERAGE	4.9619	1.8159

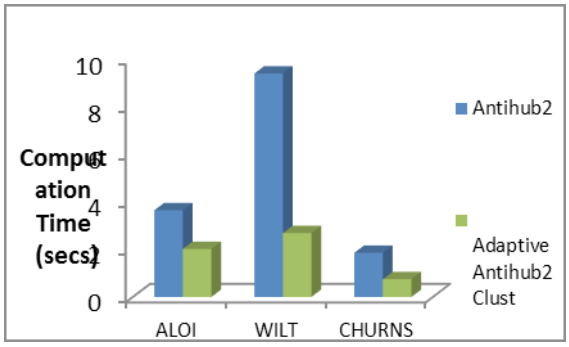

Figure 2. The computation time of algorithms for ALOI, WILT and CHURN data sets.

It is inferred from the Figure 2, that AdaptiveAntihub2Clust overtakes the other. On the whole it is recognized that AdaptiveAntihub2Clust has considerable reduction in computation time.

Table 2. The performance accuracy of algorithms

	k Value	Antihub2	AdaptiveAntihub2 Clust
ALOI	10	0.7703	0.809
	50	0.7577	0.8069
	100	0.7586	0.8204
	120	0.7595	0.806
	Average	0.761525	0.810575
WILT	10	0.9813	0.9919
	50	0.9829	0.9921
	100	0.9827	0.9917
	120	0.9829	0.9923
	Average	0.98245	0.992
CHURN	10	0.9904	1
	50	0.9988	1
	100	0.9418	1
	120	0.9328	1
	Average	0.96595	1

Table 2 represents the accuracy of all algorithms. There is a significant growth in accuracy in AdaptiveAntihub2Clust than the other.

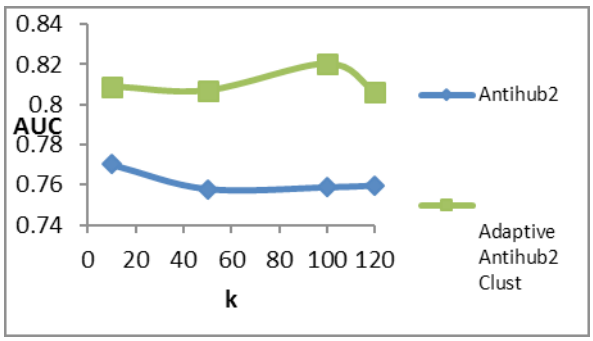

Figure 3. The performance accuracy of algorithms for ALOI dataset.

The presentation of accuracy for AdaptiveAntihub2Clust for aloi indicated in Figure 3 denotes the improvement. By seeing Table 1 for computation time for aloi, AdaptiveAntihub2Clust has reduced 44.65% of its computation time in an average with 6.05% increase in accuracy than Antihub2 in ALOI set.

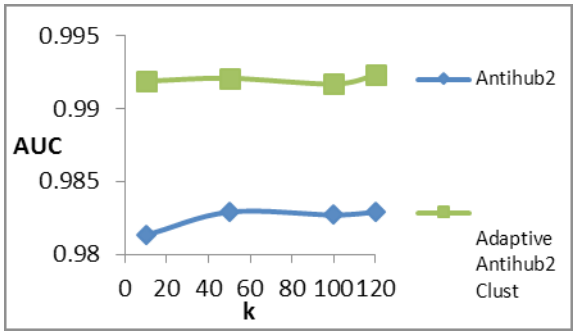

Figure 4. The performance accuracy of algorithms for WILT dataset.

Performance accuracy is indicated in Figure 4 for wilt and it brings out that AdaptiveAntihub2Clust has a considerable decrement in computation time and got reduced 71.41 % of its time with significant increase in accuracy in AdaptiveAntihub2Clust than Antihub2 algorithm in WILT dataset.

Accuracy of AdaptiveAntihub2Clust is expressed in Figure 5 for churn set. Here AdaptiveAntihub2Clust has a considerable increase in accuracy than the other with 51.09% of reduction in computation time.

5. Conclusion

This paper offers a Clustered adaptive antihub method which is put into Antihub2 for finding inconsistent objects more specifically to lessen the time elapsed. On examining the evaluation results of Antihub2 and AdaptiveAntihub2Clust, AdaptiveAntihub2Clust results well in the reduction of computation time and improvement of accuracy. So the analysis states that newly projected approach can be utilized for distinguishing outlier and it overtakes in performance than the other.

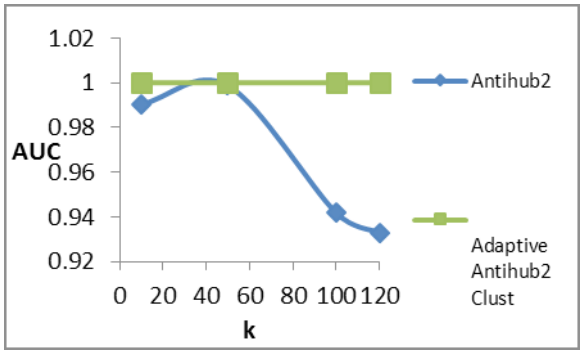

Figure 5. The performance accuracy of algorithms for CHURN dataset.

6. References

1. Aggarwal CC, Yu PS. Outlier detection for high dimensional data. ACM Sigmod Record. 2001 May 1; 30(2):37–46.
2. Chandola V, Banerjee A, Kumar V. Anomaly detection: a survey. ACM computing surveys (CSUR). 2009 Jul 1; 41(3):15.
3. Devi RL, Amalraj R. Hubness in unsupervised outlier detection techniques for high dimensional data – a survey. International Journal of Computer Applications Technology and Research. 2015; 4(11):797–801.
4. Radovanovic M, Nanopoulos A, Ivanovic M. Reverse nearest neighbors in unsupervised distance-based outlier detection. IEEE Transactions on Knowledge and Data Engineering. 2015 May 1; 27(5):1369–82.
5. Radovanović M, Nanopoulos A, Ivanović M. Hubs in space: popular nearest neighbors in high-dimensional data. The Journal of Machine Learning Research. 2010 Mar 1; 11:2487–531.
6. Ramaswamy S, Rastogi R, Shim K. Efficient algorithms for mining outliers from large data sets. ACM SIGMOD Record 2000. 2000 May 16; 29(2):427–38.
7. Breunig MM, Kriegel HP, Ng RT, Sander J. LOF: identifying density-based local outliers. ACM Sigmod Record 2000. 2000 May 16; 29(2):93–104.
8. Zhang K, Hutter M, Jin H. A new local distance-based outlier detection approach for scattered real-world data. Advances in Knowledge Discovery and Data Mining 2009; Berlin Heidelberg: Springer; Apr 27. p. 813–22.

9. Kriegel HP, Kröger P, Schubert E, Zimek A. LoOP: local outlier probabilities. Proceedings of the 18th ACM Conference on Information and Knowledge Management; ACM; 2009 Nov 2. p. 1649–52.
10. Amer M, Abdennadher S. Comparison of unsupervised anomaly detection techniques. [Bachelor's Thesis]. 2011 Sep 20.
11. Amer M, Goldstein M. Nearest-neighbor and clustering based anomaly detection algorithms for rapidminer. Proceedings of the 3rd RapidMiner Community Meeting and Conference (RCOMM 2012); 2012. p. 1–12.
12. Jain AK. Data clustering: 50 years beyond K-means. Pattern Recognition Letters. 2010 Jun 1; 31(8):651–66.
13. Wu X, Kumar V, Quinlan JR, Ghosh J, Yang Q, Motoda H, McLachlan GJ, Ng A, Liu B, Philip SY, Zhou ZH. Top 10 algorithms in data mining. Knowledge and Information Systems. 2008 Jan 1; 14(1):1–37.
14. Yoon KA, Kwon OS, Bae DH. An approach to outlier detection of software measurement data using the k-means clustering method. Empirical Software Engineering and Measurement, 2007. ESEM 2007 First International Symposium; IEEE; 2007 Sep 20. p. 443–5.
15. Münz G, Li S, Carle G. Traffic anomaly detection using k-means clustering. GI/ITG Workshop MMBnet. 2007 Sep.
16. Pamula R, Deka JK, Nandi S. An outlier detection method based on clustering. 2011 Second International Conference on Emerging Applications of Information Technology (EAIT); IEEE; 2011 Feb 19. p. 253–6.
17. Muniyandi AP, Rajeswari R, Rajaram R. Network anomaly detection by cascading k-Means clustering and C4. 5 decision tree algorithm. Procedia Engineering. 2012 Dec 31; 30:174–82.
18. Sharma SK, Pandey P, Tiwari SK, Sisodia MS. An improved network intrusion detection technique based on k-means clustering via Naïve bayes classification. 2012 International Conference on Advances in Engineering, Science and Management (ICAESM); IEEE; 2012 Mar 30. p. 417–22.
19. Tomašev N, Brehar R, Mladenić D, Nedevschi S. The influence of hubness on nearest-neighbor methods in object recognition. 2011 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP); 2011 Aug 25. p. 367–74.
20. Devi RL, Amalraj R. An efficient unsupervised adaptive antihub technique for outlier detection in high dimensional data. The International Journal of Engineering and Science (IJES). 2015; 4(11):70–7.
21. Loureiro A, Torgo L, Soares C. Outlier detection using clustering methods: a data cleaning application. Proceedings of KDNet Symposium on Knowledge-based Systems for the Public Sector. Bonn, Germany: 2004.