European Journal of Applied Sciences 9 (1): 44-49, 2017
ISSN 2079-2077

DOI: 10.5829/idosi.ejas.2017.44.49

Outlier Detection on High Dimensional Data Using RNN

R. Rohini and M. Blessa Binolin Pepsi

Department of Information Technology,
Mepco Schlenk Engineering College, Sivakasi, Tamilnadu, India

Abstract: *Background*: Outlier detection is an important factor in data mining since it is used in various real time applications. Outlier is an extreme points that are not related to any of the class. Dealing with dimensions is the great challenge, due to "curse of dimensionality", for effective outlier detection. In a high dimensional data space, it is difficult to detect most related points and most unrelated points. Outlier is the most unrelated points and in the high dimensional data all data points seemed to be a good outlier, which is a great challenge to identify. *Objective*: Reverse nearest neighbor technique is used to determine the occurrence of the points in K-nearest neighbor list. The most frequently occurring points are referred as hub points and rarely occurring points are referred as anti-hub points. *Results*: Unsupervised learning helps to find the clear outliers, throughout this paper we deal with both synthetic data and real data, to detect the clear outliers. Based on analysis, the proposed work shows that RNN-distance based similarity provides higher percentile score to detect outliers when compared to basic KNN approach and ABOD method. In addition ID3 algorithm is used to improve the detection of outlier points. This ID3 algorithm works good for all real datasets as well as synthetic datasets.

Key words: Outlier · High-dimensional data · Reverse nearest neighbor · Hub · Antihub

INTRODUCTION

Outlier means the points in extreme level i.e., the points behave abnormally. It is an anomaly detection where the extreme points refer to. This outlier detection is widely applied in practice since there is no formal mathematical definition detection. Outlier detection can be done in three methods like supervised, semi-supervised and unsupervised. But unsupervised method is most widely applied is because it is not possible for all data points would contain labels. Unsupervised method concentrate mainly on distance due to "curse of dimensionality". As dimensionality increases all points in data are seemed to be a good outlier. Hence distance concentration is done here for dealing high dimensional data. It is difficult to understand how increase in dimension impacts the outlier detection. Outlier is the noisy data that leads to unwanted data points to enter in, hence it is necessary to avoid such data and among the detection unsupervised method plays a wide role in medical research, military applications etc.,. The "curse of dimensionality" is the accepted fact that all points are identified as an equal outlier in a high-dimensional data.

Reverse nearest neighbor is proposed in this paper to identify some points clearly that behave as outlier. This technique uses the hotness phenomenon to identify the clear outlier. Since the anomaly detection helps to identify the intruders it is necessary to find the related and unrelated points. No unrelated points can come under related points. So here hubness phenomenon helps to look deeply the points occurring often and the points that occur rarely and is based on local and global. This affects the reverse nearest counts i.e. k-occurrences, the number of times point 'x' occurs among the k-nearest neighbor of other points. Some points occur frequently in the KNN list called hub points and some points occur rarely in the KNN list called antihub points. This is refered as the infrequent members. This antihub points help in detecting the outlier by a correlation factor. Here how the emergences of antihub points occur and how it leads to detection of outlier are considered. In low dimensional data the antihub points are easy to identify and in high dimensional data it is identified by increasing the neighborhood size almost equal to the size of the data. This increase in neighbor size leads to effective detection of outlier and avoid the noisy data. Based on

Corresponding Author: R. Rohini, Department of Information Technology, Mepco Schlenk Engineering College, Sivakasi, Tamilnadu, India.

the relation between outlier and antihub in both high dimensional and low dimensional data ODIN method was considered.

The occurrences of k are not regular in all cases. It differs based on the dimensions. It is well suited for low dimensional data and for high dimensional data is not suited well. Hence reverse nearest neighbor technique is used.

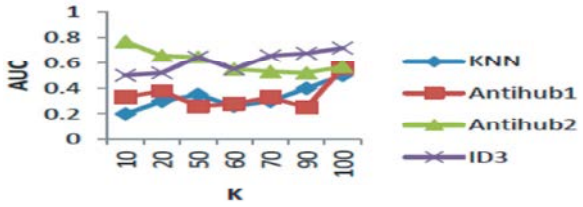

Fig. 1: Outlier for NBA-ALL STAR dataset

Related Work: In this section, we describe the existing work related to the outlier detection.

Chandola, Banerjee and kumar [1] proposd a survey for anomaly detection and shown that unsupervised learning technique is applied most widely since there is no more labeled data available. As internet grows and many data are uploaded daily with unlabelled data this technique is reduced. They also proved that time complexity is reduced by using this technique.

Hautamaki, Karkkainen and Franti [2] proposed a method of outlier detection based on k-nearest neighbor graph. This method performs the construction of kNN graph and found the anomaly point with the node with less number of in-degree i.e. with less number of nearest node. Threshold is fixed based on the analysis of all the nodes and detection of outlier is almost accurate in this method.

Aggarwal and Yu [3] describes a method of outlier detection for high dimensional data. This shows how the increase in dimension impacts the outlier detection. As the dimensionality increases the outlier points are skewed for each case [15]. This paper gives the survey of how high dimensional data works for finding distance in unsupervised learning. Skewness is measured for the outlier score so that variation is viewed clearly to find the change in low and high dimension data.

Kriegal, Schubert and Zimek [4] proposed a method called Angle Based Outlier Detection in which all data points are considered as node and based on the grid format the angle threshold is fixed to find the most neighborhood points among all the data points. This threshold differs for each dataset mainly based on the dimensionality.

Zimek, Kriegal and Schubert [5] also proposed a survey of how unsupervised learning helps to find outlier in high dimensional data. As internet grows in uploading lot of data without class label it is difficult to label all the data points. Clustering technique helps to find the anomaly for unlabelled data.

The result of the related work illustrates that dealing high dimensional data is most challenging factor for detecting outlier. The KNN and ODIN methods help to develop the proposed work for finding the effective outlier points. The analysis shows that reverse nearest neighbor technique is necessary for the unsupervised learning. The accuracy of outlier is high and here it is dealed with real real datasets to show the accuracy in outlier. The skewness is also measured for the outlier score for both high as well as and low dimensional data.

System Description: The outlier detection is done by the following methodologies for many real datasets.

- Euclidean distance
- Calculating $N_k(x)$
- Outlier score and performance analysis

The Figure 1 shows that the overview of the proposed system. Pre-processing is done for eliminating missing values and making the data ready for the proposed work is done. Distance calculation is by default is a Euclidean Distance. This distance is for finding neighbor among the data points.

The distance is an input for knn classification. The knn classification results in the list k-nearest neighbor for all points in the dataset. Among the list of k-nearest neighbor we find some points occur rarely and some points occur frequently known as antihub points and hub points respectively. Then finally outlier score is calculated and the outlier points are identified.

Implementation: This briefly discusses about the implementation methodology of the proposed work. The proposed system implemented with the following modules

- Euclidean distance
- KNN classification
- Calculating $N_k(x)$
- Determining Outlier Score

Euclidean Distance: The distance between two points of the xy-plane can be found using the distance formula. The distance between $(x1, y1)$ and $(x2, y2)$ is given by:

$$d = \sqrt{(x1-x2)^2 + (y1-y2)^2} \quad (1)$$

Similarly for high dimensional data considering all the dimensions the distance value is calculated.

KNN Classification: Given point *x* the *k* most similar points to *x* is found i.e. k nearest neighbor. This is calculated based on distance calculated in equation 1. If the distance is less it is considered

As a neighbor point similarly k nearest for each point is calculated.

Classification Steps

Step 1: Find the class label for each of the *k* neighbor.

Step 2: Use a voting or weighted voting approach to determine the majority class among the neighbor.

Step 3: Weighted voting means the closest neighbors count

Step 4: Assign the majority class label to *x*

This classification results in the list of K nearest neighbor for all points in the dataset. This helps in finding the hub points and antihub points.

Calculating $N_k(x)$: Let $D \in R^d$ be a finite set of n points. For point $x \in D$ and a given distance or similarity measure, the number of k occurrences, denoted $N_k(x)$, is the number of times x occurs among the k nearest neighbors of all other points in D.

For $q \in (0,1)$ hubs are the points $x \in D$ with highest value of $N_k(x)$.

For $q \in (0,1)$ antihubs are the points $x \in D$ with lowest value of $N_k(x)$.

In the list of k-nearest neighbor of all points in the dataset, the number of occurrences of each points is calculated. Here considering the dimensions, the k value is set almost to the size of data. Otherwise the hub points will be very less and antihub points will be high. To conclude, in high dimensions every data point is far away from the mean data and is difficult to find the outlier points. If a constant threshold value is used then it will erroneously classify every data point as an outlier. However, statistical theory used defines a reliable rule to detect outliers, regardless of the dimension of the data. This is one of the great challenges in this proposed work. The threshold value only decides the antihub points that lead to the outliers. So this is very carefully fixed. Thus the varying threshold helps to find more accurate outlier and is effectively done here [6-10].

Determining Outlier Score: The outlier score is determined by the following proposed algorithm.

Algorithm 1: $Antihub_{dist}$

Step 1: $t = N_k(x_i)$ computed with respect to distance (2)

step 2 : $s_i = f(t)$ (3)

where $f(t) = [1 / (1+ N_k(x_i))]$ is a monotone function. The disadvantage of the antihub algorithm is contributing the two factor. They are hubness property and inherent property. In order to add more discrimination one approach could be to raise k, possibly to some value comparable with n.

But the approach raises two main factors:

- With increasing k the notion of outlierness moves from local to global, thus if local outliers are concentrated in this methodology.
- K values comparable with n raise issues with computational complexity.

Algorithm 2: $Antihub^2$

$AntiHub^2$ refines outlier scores produced by the AntiHub method by also considering the N_k scores of the neighbors of x, in addition to $N_k(x)$ itself. For each point x, $AntiHub^2$ proportionally adds $(1-\alpha)$. $N_k(x)$ to α times the sum of N_k scores of the k-nearest neighbor of x where $\alpha \in [0,1]$.

Step 1: z=Antihub dist

This is the outlier score of the previous step

step 2 : $yi = \Sigma j \, ^a N \, Ndist(k,i) \, zj,$ (4)

where Ndist (k,i) is the set of indices of k nearest

neighbors of xi and initialize temp=0

step 3 : $Si : = (1- \alpha). ai + \alpha. yi$ (5)

Here α value is chosen between (0,1)

Step 4 : calculation of discScore

dis : = discScore (S,p) (6)

This function returns the value u/np where u is the number of unique points in yi and n is the number data.

Step 5: If dis > temp then t is set to S and disc is set to temp

Step 6: Outlier Score Calculation: si = f (t)

where f (t) = [1 / (1+ No (xi))] is a monotone function.

Thus anithub[2] algorithm refines the outlier score. This score proves that the accuracy in determining the outlier with various values of k is high. This proposed algorithm reduces the time complexity and manages the whole system contributions in a good formation. The different datasets are used and results are taken for the analysis. The dataset varies with the dimension which is concentrated more here. Thus data points are mostly similar to outliers and this method went deeply in finding the data points with that are considered as outliers. Time complexity is expensive since the data points are with high dimensions. This algorithm is well suited for both low-dimensions and high dimensional values. The ID3 based approach also gives equally good results for finding the outlier points irrespective from the dimensions. This approach gives the good sore similar to the antihub[2] method which suits for the both synthetic data and real data.

Result Analysis:

Table I: Real Dataset Used in the Experiment

			Sn10		Outlier%	
Data set	n	D	Antihub[2]	ID3	Antihub[2]	ID3
Aloe	1536	18	0.158	0.091	7.53	7.63
Churn	732	17	0.485	0.389	11.925	12.3
KDD	1234	38	0.792	0.566	9.34	9.66
Mammo Graphy	738	6	0.244	0.136	4.07	4.57
Nba-Allsta R	452	14	0.28	0.153	19.394	19.454
Thyroid Sick	1150	24	0.496	0.342	7.36	8.66

This table shows that mammography dataset is the low dimensional data and KDD dataset is the high dimensional data. The outlier detection for the proposed method suits well for all the above cases with the greater accuracy comparing KNN algorithm. The proposed method also reduces the time complexity since there is no need of assigning labels for all data points [11-18]. Considering the binary classification in KNN it is necessary to give class label for all points and it increases the time complexity. The advantage is that skewness value differs for each dimension as the dimension plays the major role here.Real dataset is used here for better analysis by considering all the challenges and the outliers are obtained as follows.

Table I describes the skewness and outlier for different dataset for antihub[2] and ID3 approach. The skewness value is found low and outlier percentage is high in ID3 method. The data points found as outlier are almost similar to antihub[2].

Table II: Strong Outlier for Nba Allstar Data in Antihub[1]

		Outlier	
Name	Year	Antihub[2]	ID3
Kareem Abdul Jabbar	1979	Yes	Yes
Shawn Marion	1984	No	Yes
Dimton jack	1977	No	Yes
Vilowon Kae	2002	Yes	Yes

This table shows the strong outliers that are detected in the nba-allstar dataset. These points are considered as a noisy data or the unwanted data i.e. anomaly. The points can be removed and the data points can be made available for further use.

Table III: Strong Outlier for Nba Allstar Data in Antihub[2]

		Oulier	
Name	Year	Antihub[2]	ID3
Kareem Abdul – Jabbar	1979	Yes	Yes
Shawn Marian	2004	No	Yes
Dimton jack	1986	Yes	Yes
Vilowon Kae	1978	Yes	Yes

This table gives the refinement of the outlier score in Antihub[2] algorithm. This makes the correctness of the score and shows the there is no wide change in the outlier points. Some of the points are misclassified as good points in antihub[1] are correctly classified in this antihub[2] algorithm. Thus it gives the refinement of outlier score. Since nba-allstar dataset works well for all the data algorithm this is shown in brief. The effective reduction of time complexity is achieved by avoiding the class labels of using unsupervised learning methodology.

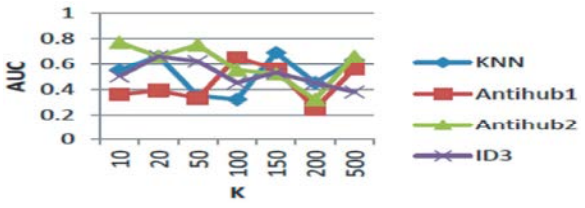

Fig. 3: Outlier for ALOI Dataset

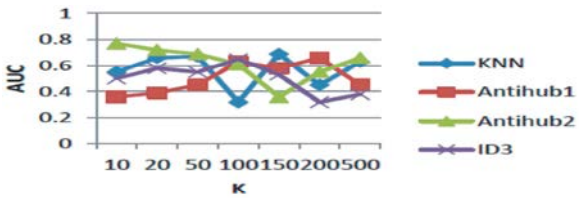

Fig. 4: Outlier for WILT Dataset

This figure describes the outlier detection for the three methods where antihub[2] method finds the better result. It is to be noted that skewness is reduced for the antihub[2] method and ID3 method, so the accuracy will also be high correspondingly. One of the challenging is fixing threshold value and it is done on the analysis. Here k is the number of neighbor value and AUC is the Area under curve is by f(t) used for finding the outlier score. This is related to the detection of points where existence of anomaly are identified inn higher accuracy rate comparing to KNN, ABOD and ODIN methods.

CONCLUSION

This section concludes with the discussion of obtained result and future enhancement. The outlier detection is one of the most important works in data mining. This was detected effectively in the proposed algorithm. Antihub[2] algorithm is more efficient than knn and antihub[1] algorithms. Some points are seemed to be good outliers in the antihub[1] algorithm this is due to the curse of dimensionality. It is overcome in the antihub[2] algorithm in a successful manner. The class label was not present in the unsupervised learning and therefore it may loss some accuracy. The time complexity is less compared to the supervised leaning methodologies. This work finds the great support in testing all the real dataset and it continues the experiment in an high accuracy that leads to a great success of this experiment. The high dimensional data leads all points to be a good outlier. To overcome this challenge they introduced hubness phenomenon and that is well suited for the proposed algorithm. The existence of hubs and antihubs in high-dimensional data is relevant to machine-learning techniques from various families: supervised, semi-supervised, as well as unsupervised. Based on the analysis, they formulated the AntiHub method for unsupervised outlier detection discussed its properties and proposed a derived method which improves discrimination between scores. The main hope is that this article clarifies the picture of the interplay between the types of outliers and properties of data, filling a gap in understanding which may have so far hindered the widespread use of reverse neighbor methods in unsupervised outlier detection. In future this technique can be applied for Image Mining.

REFERENCES

1. Chandola, V., A. Banerjee and V. Kumar, 2009. Anomaly detection: Asurvey. ACM Comput. Survey, 41(3): 15.
2. Hautamaki, V., I. Karkkainen and P. Franti, 2004. Outlier detection using k-nearest neighbor graph, Proc 17th Int. Conf. Pattern Recognit., 3: 430-433.
3. Aggarwal, C.C. and P.S. Yu, 2001. Outlier detection for high dimensional data. Proc. 27th ACM SIGMOD Int. Conf. Manage. Data, pp: 37-46.
4. Kriegel, H.P., M. Schubert and A. Zimek, 2008. Angle-based outlier detection in high-dimensional data. Proc 14th ACM SIGKDD Int. Conf. Knowledge. Discovery Data Mining, pp: 444-452.
5. Zimek, A., E. Schubert and H.P. Kriegel, 2012. A survey on unsupervised outlier detection in high-dimensional numerical data. Statist. Anal. Data Mining, 5(5): 363-387.
6. Beyer, K.S., J. Goldstein, R. Ramakrishnan and U. Shaft, 1999. When is "nearest neighbor" meaningful?. Proc. 7th Int. Conf. Database Theory, pp: 217-235.
7. Breunig, M.M., H.P. Kriegel, R.T. Ng and J. Sander, 2000. LOF: Identifying density-based local outliers. SIGMOD Rec., 29(2): 93-104.
8. Papadimitriou, S., H. Kitagawa, P. Gibbons and C. Faloutsos, 2003. LOCI: Fast outlier detection using the local correlation integral. Proc 19th IEEE Int. Conf. Data Eng, pp: 315-326.
9. Zhang, K., M. Hutter and H. Jin, 2009. A new local distance-based outlier detection approach for scattered real-world data. In Proc 13th Pacific-Asia Conf. Knowl, Discovery Data Mining, pp: 813-822.
10. Kriegel, H.P., P. Kreoger, E. Schubert and A. Zimek, 2009. LoOP: Local outlier probabilities. in Proc 18th ACM Conf. Inform. Knowl. Manage., pp: 1649-1652.
11. Houle, M.E., H.P. Kriegel, P. Kreoger, E. Schubert and A. Zimek, 2010. Can shared-neighbor distances defeat the curse of dimensionality? in Proc 22nd Int. Conf. Sci. Statist.DatabaseManage., pp: 482-500.
12. Singh A., H. Ferhatosmano and A. Saman Tosun, 2003. High dimensional reverse nearest neighbor queries. Proc 12th ACM Conf. Inform. Knowl. Manage., pp: 91-98.

13. Tao, Y., M.L. Yiu and N. Mamoulis, 2006. Reverse nearest neighbor search in metric spaces, IEEE Trans. Knowl. Data Eng., 18(9): 1239-1252.
14. Lijun, C., L. Xiyin, Z. Tiejun, Z. Zhongping and L. Aiyong, 2010. A data stream outlier delection algorithm based on reverse k nearest neighbors. Proc. 3rd Int. Symp. Comput. Intell. Des., pp: 236-239.
15. Jin, W., A.K.H. Tung, J. Han and W. Wang, 2006. Ranking outliers using symmetric neighborhood relationship. in Proc 10th Pacific-Asia Conf. Adv. Knowl. Discovery Data Mining, pp: 577-593.
16. Kriegel, H.P., P. Kreoger, E. Schubert and A. Zimek, 2011. Interpreting and unifying outlier scores. Proc 11th SIAM Int. Conf. DataMining, pp: 13-24.
17. Erdos, P. and A. Renyi, 1959. On random graphs,. Publication Math-Debrecen, 6: 290-297.
18. Bache, K. and M. Lichman, 2014. UCI machine learning repository: http://archive.ics.uci.edu/ml.