

A Spectral Clustering Based Outlier Detection Technique

Yuan Wang¹, Xiaochun Wang^{1(✉)}, and Xia Li Wang²

¹ School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China
wy734174981@gmail.com, xiaocchunwang@mail.xjtu.edu.cn

² School of Information Engineering, Changan Univeristy, Xi'an 710061, China
xlwang@chd.edu.cn

Abstract. Outlier detection shows its increasingly high practical value in many application areas such as intrusion detection, fraud detection, discovery of criminal activities in electronic commerce and so on. Many techniques have been developed for outlier detection, including distribution-based outlier detection algorithm, depth-based outlier detection algorithm, distance-based outlier detection algorithm, density-based outlier detection algorithm and clustering-based outlier detection. Spectral clustering receives much attention as a competitive clustering algorithms emerging in recent years. However, it is not very well scalable to modern large datasets. To partially circumvent this drawback, in this paper, we propose a new outlier detection method inspired by spectral clustering. Our algorithm combines the concept of kNN and spectral clustering techniques to obtain the abnormal data as outliers by using the information of eigenvalues and eigenvectors statistically in the feature space. We compare the performance of our methods with distance-based outlier detection methods and density-based outlier detection methods. Experimental results show the effectiveness of our algorithm for identifying outliers.

Keywords: Outlier detection · Distance-based outlier detection · Density-based outlier detection · Spectral clustering · Eigenvalues

1 Introduction

With the rapid development of information technology, people have been able to easily access and store large amounts of information from the real world. However, how to find important and useful information from these massive and high-dimensional data has become an urgent problem. Therefore, data mining and database technologies come into being consequently.

Outlier detection is an important data mining technique, which focuses on discovering the small portion of data objects in a data set that, being inconsistent with the conventional pattern of the majority of the data set, however, may imply important information. Hawkins proposed a relatively widely accepted definition: “Outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism.” [1] Currently, outlier detection has important applications in areas like intrusion detection [2,3], fraud detection [4], etc..

In recent years, there have been a broad range of definitions for outliers and researchers have developed many types of algorithms for outlier detection, mainly including distribution-based outlier detection algorithm, depth-based outlier detection algorithm, distance-based outlier detection algorithm, density-based outlier detection algorithm and clustering-based outlier detection [5,6]. However, none of them has been proved to be completely applicable to every situation. Methods based on the statistical distribution are firstly presented using the classic statistical approaches with the assumption that the database corresponds to a given distributional model. However, the model is usually not known a priori for modern sophisticated large databases. As a result, they have a limited number of applications. Being an improvement, depth-based outlier detection methods assign a depth value to each data object and regard the data objects in the shallow layers to be more likely to be outliers than those in the deep layers. Unfortunately, these methods suffer high computational complexity for more than a few dimensions. Distance-based methods, also known as adjacency-based methods, believe that data objects are outliers if they are far away from the majority of data points and address more globally-oriented outliers in the database [7]. Density-based methods usually assign each data object a measure of outlier degree as the classic LOF algorithm did and then regard those data objects which possess the largest outlier degrees as outliers [8]. In comparison to distance-based methods, these methods address more locally-oriented outliers. Finally, clustering-based methods obtain outliers as a by-product and regard the outliers as the data items that reside in the smallest clusters [9].

Recently, spectral clustering has been widely applied to pattern recognition and data mining because it can obtain global optimal solution on sample spaces of arbitrary shape [10,11,12]. Meanwhile, since it is only related to the number of data points, not the dimensionality, spectral clustering can help solve the curse of dimensionality suffered by distance-based methods and density-based methods on high-dimensional data space. In this paper, we propose a spectral clustering based outlier mining approach, which, when compared with distance-based methods and density-based methods on some standard test datasets, manifests its effectiveness and efficiency.

The rest of the paper is organized as follows. In section 2, we present some preliminaries of spectral clustering. In section 3, the proposed spectral clustering based outlier mining approach is introduced. In section 4, we present the results of our experiments conducted to evaluate the performance of our algorithm. Finally, the conclusions are given in section 5.

2 Preliminaries

2.1 Spectral Clustering

Clustering is the process to divide points in a data set into a number of categories of clusters so that the similarity between two points within the same cluster and the dissimilarity between two points belonging to respectively two different clusters is as high as possible. Cluster analysis plays an important part in data mining. Spectral clustering receives much attention as a competitive clustering algorithms emerging in

recent years, which is mainly applied to image segmentation. Basically, possessing numerous advantages, spectral clustering treats points in a dataset as the vertices of a weighted undirected graph and the similarity between two vertexes as the weights of edges in the graph, and converts the problem of clustering into an optimal partitioning problem of a graph. The goal is to find a graph partitioning methodology to make the weights of edges connecting two different sub-graphs as large as possible and the weights of edges within a sub-graph as small as possible. The most common graph division criteria proposed include Mini Cut [13], Normalized Cut [14,15] and Ratio Cut [16]. Mini Cut produces a good result on image segmentation but is prone to skew-segmentation, while Normalized Cut and Ratio Cut take both the minimum sum of the weights of the cutting edges and the balance of division into account.

In this paper, we utilize the k -way Normalized Cut partitioning approach, for a given data set, which contains n points, x_1, x_2, \dots, x_n , the goal of clustering is to divide this n points into k clusters so that the similarities between two points belonging to a same cluster are maximized and those between two points belonging to different clusters are minimized. Treating data point x_i in the dataset as a vertex v_i in a graph and the similarity between x_i and x_j , W_{ij} , as the weight of the edge connecting x_i and x_j , and we obtain a weighted undirected graph, $G = (V, E)$ with V the set of all the vertexes and E the set of all the edges in the graph G . Suppose that the multi-way spectral clustering partitions the graph G into k sub-graphs, A_1, A_2, \dots, A_k , then the target function of Normalized Cut ($Ncut$) that makes the optimal partitioning can be described as:

$$Ncut(A_1, A_2, \dots, A_k) = \sum_{i=1}^k \frac{cut(A_i, \bar{A}_i)}{vol(A_i)} \quad (1)$$

$$\text{where } cut(A, B) = \sum_{i \in A, j \in B} W_{ij}, \bar{A}_i = \{v_p \mid v_p \in V, v_p \notin A_i\}, vol(A_i) = \sum_{p \in A_i, q \in V} W_{pq}.$$

After mathematic transformations, the solution for the above $Ncut$ problem can be finally converted to a problem of finding eigenvalues (and eigenvectors) of a Laplacian matrix and the smallest series of eigenvalues corresponds to the optimal partitioning of the graph. Thus the discrete clustering problem becomes to find the eigenvectors on a contiguous data space.

In the general frame of the spectral clustering, the following three Laplacian matrices are commonly used,

$$L = D - W \quad (2)$$

$$L_{sys} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \quad (3)$$

$$L_{rw} = D^{-1} L = I - D^{-1} W \quad (4)$$

where W is the similarity matrix, D is the corresponding diagonal matrix, $D_{ij} = \sum_{j=1}^N W_{ij}$, and I is the identity matrix. In this paper, we adopt the second form of Laplacian matrix, and notice that $L_{\text{sys}} = I - D^{-1/2} W D^{-1/2}$, hence the sum of eigenvalues of $D^{-1/2} W D^{-1/2}$ and the corresponding eigenvalues of L_{sys} are one and the corresponding eigenvectors are equal. Therefore, finding the eigenvectors of $D^{-1/2} W D^{-1/2}$ which correspond to the k largest eigenvalues is equivalent to finding the eigenvectors of L_{sys} which corresponds to the k smallest eigenvalues. With these ideas in mind, the general frame of spectral clustering works as following,

1. Given a set of data points, x_1, x_2, \dots, x_n , calculate the similarity between each two points according to a certain kind of similarity definition (e.g., the Gaussian kernel function) and construct the similarity matrix W ;
2. Compute the corresponding Laplacian matrix L_{sys} , and find its k eigenvectors corresponding to the k smallest eigenvalues and use these k eigenvectors as columns to construct a feature matrix in a k -dimensional space, $H \in R^{n \times k}$;
3. Treat each row of feature matrix, H , as a data point in k -dimensional space and perform K -means clustering on these points. Each original d -dimensional data point is assigned the same cluster number as the k -dimensional feature vector of the corresponding row in the feature matrix H .

2.2 An Enhancement for Spectral Clustering

From the general framework of spectral clustering given in the previous subsection, it can be seen a similarity definition should first be provided to establish the similarity matrix. The most commonly used one is the Gaussian kernel function, which has the following form,

$$W_{ij} = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}} \quad (5)$$

where $\|x_i - x_j\|$ denotes the Euclidean distance between two points x_i and x_j , and σ is the width parameter, controlling the radial function scope of the Gaussian kernel function. Obviously, the similarity is closer to 1 if two points are very close; otherwise, the similarity is closer to 0 if two points are far away from each other.

With respect to parameter σ , in the original NJW algorithm, it was proposed to perform spectral clustering respectively using several pre-set values and choose the one that yields the best clustering results. To improve the running time performance, some researchers suggested to determine σ by empirical formulas. However, all these practices require a combination of the domain knowledge and are not applied to all the situations. Therefore, researchers start to focus on how to automatically determine this parameter according to the data set itself. Under this line, Zelnik-Manor and Perona proposed the ‘‘Self-Tuning’’ algorithm [17]. Based on ‘‘Local Scale’’ idea, the algorithm constructs a self-adaptive parameter $\sigma_i = \|x_i - x_p\|$ (where x_p is the k -th nearest neighbor of x_i , usually $k = 7$ is used) for each data point using its own

neighborhood information. The similarity between two points is then replaced by the following form,

$$W_{ij} = e^{-\frac{\|x_i - x_j\|^2}{\sigma_i \sigma_j}} \quad (6)$$

Empirical evidence shows that this latter approach can address those data set with a multi-scale nature where a universal σ can not do, and isolate accurately the dense clusters embedded in a sparse background.

2.3 Outlier Detection

There are three parts of the unsupervised outlier detection literature that are related to our study: distance-based outlier detection, density-based outlier detection and clustering-based outlier detection.

Knorr and Ng proposed distance-based outlier detection methods as a good way to detect outliers residing in relatively sparse regions. Beginning with this work, various versions of distance-based outlier definition have been developed. For an example, given two integers, n and k , outliers are the data items whose distance to their k -th nearest neighbor is among top n largest ones [18], referred to as the DB-max method in the following. For another example, given two integers, n and k , outliers are the data items whose average distance to their k nearest neighbors is among top n largest ones [19,20], referred to as the DB method in the following.

Distance-based outlier detection techniques work well for detecting global outliers in simply-structured data sets that contain one or more clusters with similar density. However, for many real world data sets which have complex structures in the sense that different portions of a database can exhibit very different characteristics, they might not be able to find all interesting outliers. To deal with this situation, Breunig et al. pioneered the density-based outlier detection research by assigning to each object a degree of being an outlier, called the Local Outlier Factor (LOF), for judging the outlyingness of every object in the data set based on ratios between the local density around an object and the local density around its neighboring objects [8]. The LOF method works by first calculating the LOF for each object in the data set. Next, all the objects are ranked according to their LOF values. Finally, objects with top- n largest LOF values are marked as outliers.

A problem associated with distance-based as well as density based outlier detection algorithms is their strong sensitiveness to the setting of some parameters. The situation could be worse for the detection of outliers in high-dimensional feature space since data points cannot be visualized there. This is where clustering algorithms can be of some help. Being a very important data mining tool, the main concern of clustering algorithms is to find clusters by optimizing some criterion, such as minimizing the intra-cluster distance and maximizing the inter-cluster distance. As a by-product, data items in small groups can often be regarded as outliers (noise) that should be removed to make clustering more reliable.

3 The Proposed Spectral Clustering Based Outlier Mining Algorithm

In the above section, outlier definitions and spectral clustering are presented. In view of the curse of dimensionality problem suffered by distance-based and density-based algorithms on high-dimensional data space, while spectral clustering shields this problem by computing similarity between two data objects using Euclidean distance but suffer high computation cost and high memory storage cost. For a given dataset with N number of objects, the size of the similarity matrix used by spectral clustering is $N \times N$. When N is very large, the size of the similarity matrix will become too large to fit into the main memory.

3.1 A Simple Idea

It is generally believed outliers comprise a small portion of the whole dataset and reside in small clusters in sparse region and behave differently relative to the majority of the normal data. To take the dimension and the number of data objects in a database both into consideration for clustering based outlier detection, we propose a pre-process to find for each data point its k -nearest neighbors (kNN) and then perform a spectral clustering process on these $k+1$ data point. By this way, it is more easily to examine the abnormal behavior of a small number of data points by taking advantage of a small neighborhood of each data point and its kNN while keeping the tempo- and spatio-computational cost as low as possible. The result of this pre-processing step is N new small datasets consisting of each data point and its k nearest neighbors. Next, we perform spectral clustering on every new dataset, which results in N new sets of k -dimensional feature data in the eigen-space. For these k -dimensional feature data in the eigen-space, their smallest eigenvalue is zero for all the data. Our clustering based outlier detection algorithm is based on the observation that the values of the second smallest eigenvalues associated with outliers have lowest frequency of occurrence. However, the opposite is not true. That is, the values of the second smallest eigenvalues associated with some inliers have very low frequency of occurrence as well. From a statistical point of view, we select those points (for example 15%~20% of the total number) as outlier candidates whose corresponding second smallest eigenvalues appear least frequently. From our experience, there are some inliers among the outlier candidates. To remove inliers, we then compute the distance of each outlier candidate to its k -th nearest neighbor as its outlier index to rank the outlier candidates (i.e., the distance-based outlier score for the DB-max method). Finally, top n ranked outlier candidates with the largest outlier indices will be returned as outliers in the database.

To illustrate this observation, a synthetic 2-dimensional dataset is plotted in Fig. 1. It consists of 73 data points and, in our pre-process stage, produces 73 size-reduced mini datasets. After performing spectral clustering process on these 73 new groups of data, we plot the second smallest eigenvalues for each new data set in Fig. 2. From the plot, it can be clearly seen that six points, labeled by 68, 69, 70, 71, 72, 73, have their second smallest eigenvalues occurring with lowest frequency (i.e., one time, while other values appear more than one time) among all such eigenvalues as plotted in Fig. 2. Then back to Fig. 1, we can see they correspond to six outstanding outliers.

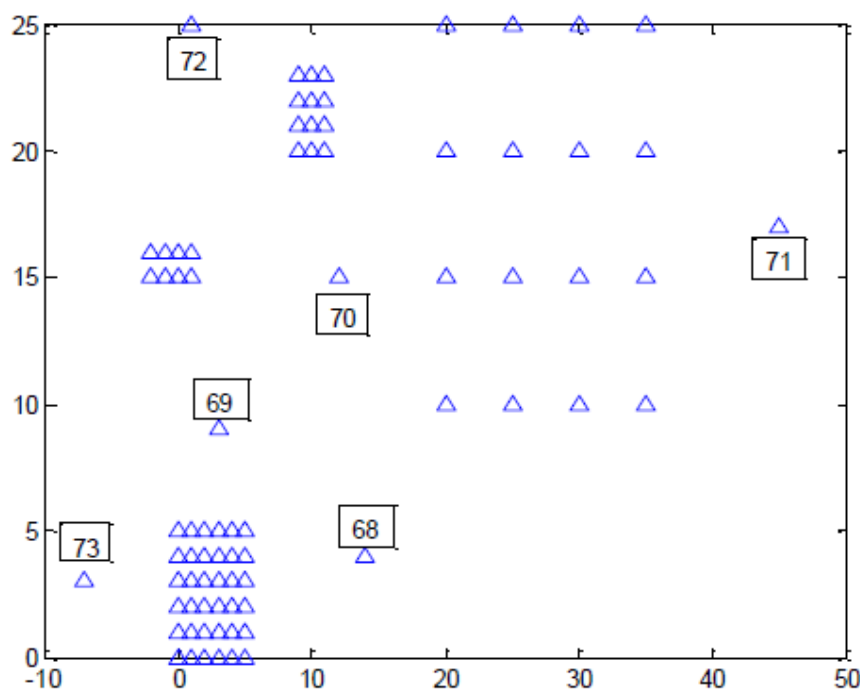


Fig. 1. Synthetic dataset for illustration of our working idea

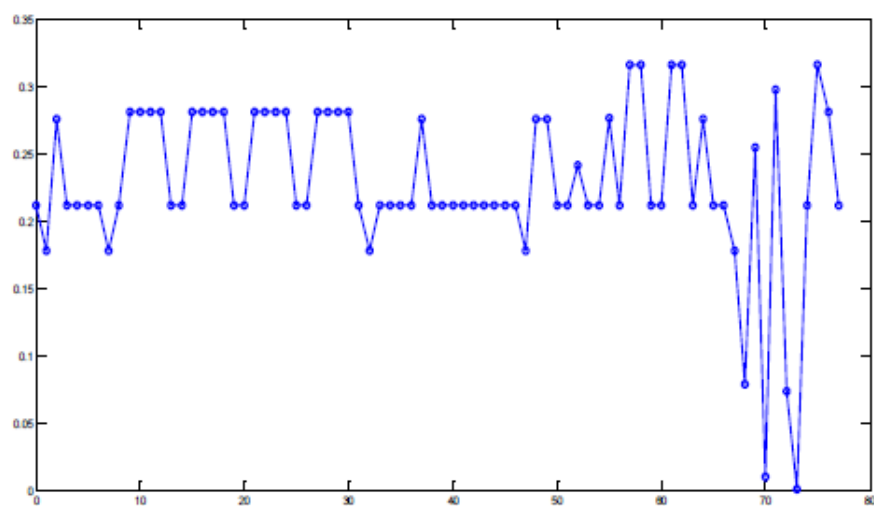


Fig. 2. The distribution of the second smallest eigenvalues of synthetic dataset

3.2 Our Spectral Clustering Based Outlier Detection Algorithm

Based on the above discussion, our spectral clustering (SC)-based outlier detection algorithm can be summarized in the following Table 1.

Table 1. A Spectral Clustering Based Outlier Detection Algorithm

Input:	S : a set of N data objects; p : a loosely estimated percentage of the number of outliers; k : the number of nearest neighbors
Output:	m : a desired number of ranked outliers;
Begin:	
1:	Compute k nearest neighbors for each data point; generate N mini datasets $\{S_1, S_2, \dots, S_N\}$ consisting of each data point and its kNN ;
2:	Perform spectral clustering on each mini dataset S_i , $1 \leq i \leq N$; collect all the corresponding second smallest eigenvalues λ_{i2} , $1 \leq i \leq N$;
3:	Add top p data objects with the least occurring frequency of the second smallest eigenvalues to outlier candidates C ;
4:	For each outlier candidate in C , calculate its distance to the k -th nearest neighbor as the outlier index;
5:	Rank all the outlier candidates in a non-increasing order according to their outlier index and return top m ones with the biggest outlier indices as the final outliers.
End	

To summarize, the numerical parameters the algorithm needs from the user include the data set, S , the loosely estimated number of outliers (i.e., the percentage of outlier candidates in the original data set), p , and the number of nearest neighbors, k .

4 Experimental Results

In this section, experiments are conducted to evaluate the performance of our proposed algorithms in comparison to those of three-state-of-the-art outlier detection methods, namely, the DB method, the DB-max method and the LOF method, on different datasets. First, we select three 2-dimensional outlier detection problems to show that our spectral clustering based algorithm can outperform classic outlier detection algorithms in the detection accuracy. And then we evaluate our algorithm on a higher dimensional real dataset with no assumptions made on the data distribution, which is downloaded from the UCI Machine Learning Repository [21], to check the technical soundness of this study. All the data sets are briefly summarized in Table 2.

We implement all the algorithms in java and perform all the experiments on a computer with AMD A6-4400M Processor 2.70GHz CPU and 4.00G RAM. The operating system running on this computer is Windows 7. In our evaluation, we focus on the outlier detection accuracy rate of these four outlier detection algorithms on different data sets. The results show that, overall, our spectral clustering based outlier detection algorithm is superior over other state-of-the-art outlier detection algorithms.

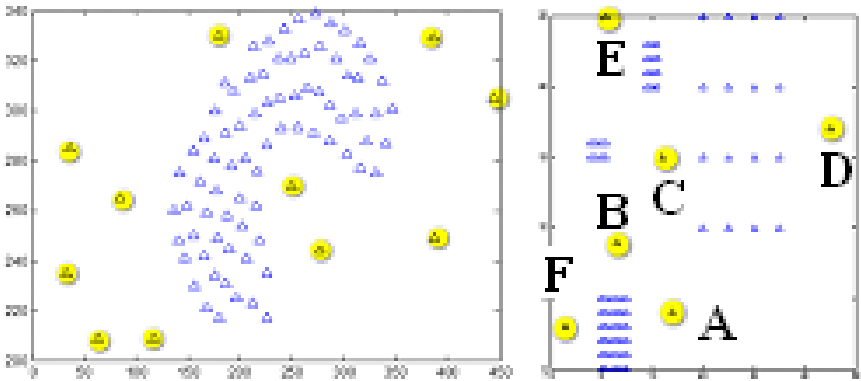
Table 2. Descriptions of all datasets

Data Name	Data Size	Dimension	# of outliers
syn_Data1	89	2	11
syn_Data2	78	2	6
lymphography	148	18	6

4.1 Performance of Our Algorithm on Synthetic Data

In this subsection, we use two synthetic datasets to show that the proposed spectral clustering based outlier detection method can effectively identify local and global outliers in various scenarios.

All two synthetic datasets, syn_Data1 and syn_Data2 are shown in Fig. 3. The first synthetic dataset, syn_Data1, consists of 89 instances, including one large uniformly distributed cluster surrounded by eleven planted easy-identified outliers denoted by yellow circles. This is a global outlier detection task. The best experimental results are obtained with the parameter k 's being determined by error and trial to be 5, 5, 5, 14 for DB, DB-max, LOF and our method, respectively and depicted in Fig. 4.


Fig. 3. Synthetic Datasets (left) syn_Data1, (right) syn_Data2

From the figure, it can be seen that, LOF method and our method can correctly detect all the outliers (denoted by red circles) while DB and DB-max methods both miss one.

syn_Data2 contains 78 instances, including five planted global outliers (A,D,E), two local outliers (B, C), and four clusters of different densities consisting of 36, 8, 12 and 16 uniformly distributed instances. To demonstrate the effectiveness of our approach in finding both global and local outliers, the same set of experiments is conducted and the best experimental results are obtained with the parameter k 's being determined by error and trial to be 5, 5, 5, 5 for DB, DB-max, LOF and our method, respectively and depicted in Fig. 5. For this case, DB and DB-max both miss the two local outliers.

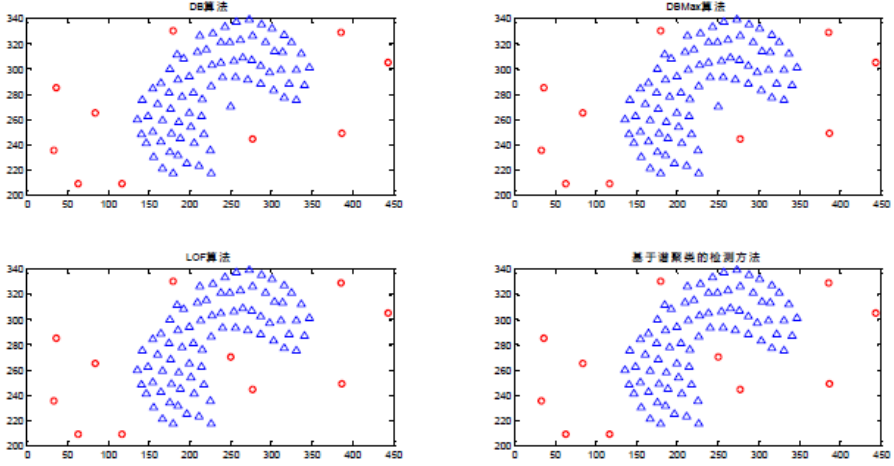


Fig. 4. Experimental results for syn_Data1 (upper left) DB method, (upper right) DB-max, (lower left) LOF method, (lower right) our method

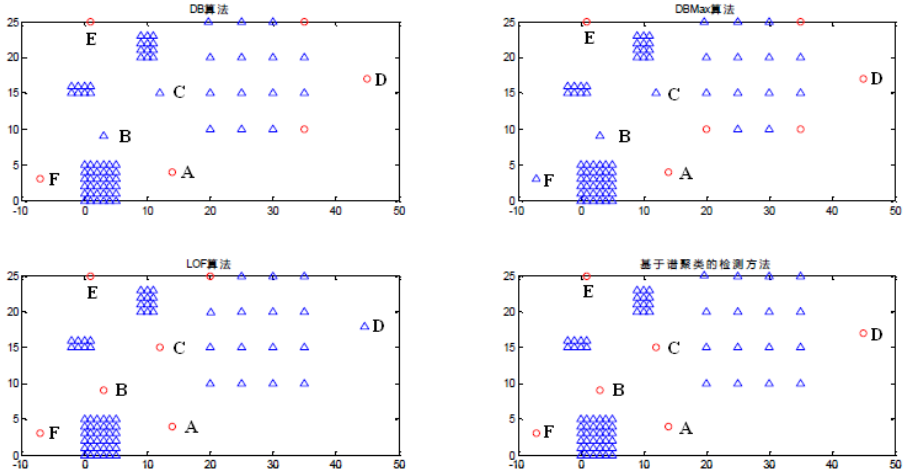


Fig. 5. Experimental results for syn_Data2 (upper left) DB method, (upper right) DB-max, (lower left) LOF method, (lower right) our method

LOF misses one global outlier. Our spectral clustering based outlier detection algorithm identifies all six outliers correctly.

4.2 Performance of Our Algorithm on Real Data

As pointed out by Aggarwal and Yu, one way to test how well the outlier detection algorithm works is to run the method on the dataset and test the percentage of points which belongs to the rare classes [22]. To evaluate the effectiveness and accuracy of

our proposed method on real data, we compare the algorithms by their performance on detecting rare classes in a real dataset, namely, lymphography, which is downloaded from UCI [21]. The lymphography dataset has 148 instances with 18 attributes and contains a total of 4 classes. Classes 2 and 3 have the largest number of instances (81 and 61, respectively). The remaining two classes have totally 6 instances (2 and 4, respectively) and are regarded as outliers (i.e., rare classes) for they are small in size. To quantitatively measure the performance of an outlier detection scheme, a metric called recall is employed here. Assuming that a dataset $D=D_o \cup D_n$ where D_o denotes the set of all outliers and D_n denotes the set of all normal data. Given any integer $m \geq 1$, if O_m denotes the set of outliers among objects in the top m positions returned by an outlier detection scheme, recall is defined as

$$recall = \frac{|O_m|}{|D_o|} \quad (7)$$

We report the corresponding detecting results of four methods in terms of recall in Table 3 with the parameter k 's being determined by error and trial to be 5, 5, 5, 8 for DB, DB-max, LOF and our method, respectively, and $m=6$. From the experimental results, it can be seen that our method performs the best.

Table 3. Experimental results for lymphography data

Dataset	DB	DB-max	LOF	Our Method
lymphography	0.33	0.50	0.50	0.67

4.3 Discussion

For an undirected, weighted graph with weight matrix, the multiplicity n of the eigenvalue 0 of the matrix equals the number of n connected components in the graph in the ideal case. The matrix has as many eigenvalues 0 as there are connected components, and the corresponding eigenvectors are the indicator vectors of the connected components. If the graph only consists of one connected component, eigenvalue 0 has multiplicity 1 and the first eigenvector is the constant vector. Therefore, if each point and its k nearest neighbors all belong to the same cluster in the ideal situation, the smallest eigenvalues for all the point's $k+1$ nearest neighbor set are zero and the rest eigenvalues are very similar if not of exactly the same values, resulting in their high frequency of appearance. However, in real situations where outliers may exist, the second smallest eigenvalues in the set of eigenvalues of the spectral clustering for each point's $k+1$ nearest neighbor set should behave (in terms of frequency of appearance) quite differently for outliers than for those of the majority of normal data points. To summarize, a low frequency of the second eigenvalue in the set of eigenvalues of the spectral clustering for each point's $k+1$ nearest neighbor set is used as an indication that this point is an outlier.

Distance-based and density-based outlier detection methods are good outlier detectors. However, they are very sensitive to parameter k and a small change in k can lead

to changes in the scores and, correspondingly, the ranking. From Section 3, we know that our method is based on a low frequency of the second eigenvalue in the set of eigenvalues of the spectral clustering for each point's $k+1$ nearest neighbor set. k is a very critical parameter of our proposed method. For a suitable value to be chosen for k , our goal is to promote eigenvalues other than the second smallest one to appear as frequently as possible. In other words, a desired k should be large enough for each point's $k+1$ nearest neighbor set to include normal points so much that these normal points dominate the neighborhood. From the experiments, we see that, for `syn_Data1` and the real data, where outlying groups have outliers more than one data point, the optimal k 's for our method are much larger than those for the other three methods, while for `syn_Data2`, where there is only one data point for each outlying group, the optimal k 's for our method is the same as those for the other three methods.

5 Conclusions

In this paper, we have proposed an effective spectral clustering based outlier detection method that can detect both global and local outliers. Traditional spectral clustering-based outlier detection algorithms have a quadratic running time complexity with data sizes and are very time- and space-consuming for modern large datasets. To partially circumvent this problem, we apply the spectral clustering process upon N mini-datasets, each consisting a data point and its k -nearest neighbors. To identify potential outliers, our algorithm first locates those data points in the eigen space whose second smallest eigenvalue appears least frequently as outlier candidates that deviate from the main patterns. Candidate outliers are then ranked based on the notion of distance-based outlier scores that are assigned to each data point. To demonstrate the utility of our proposed outlier detection mechanism, we have performed a detailed comparison of its performance with state-of-the-art distance-based and density-based outlier detection methods. Experimental results show the ability of our algorithm to rank the best candidates for being an outlier with high recall. Our study also manifests that, in reality, it is usually difficult to detect all the outliers that fit user's intuitions. Thus, it is more meaningful to incorporate our proposed outlier detection method as a component into current outlier detection framework. Finally, the size of the nearest neighborhood parameter, k , is very critical to the behavior of the proposed method. Though a qualitative analysis for how to choose it is given in the above, a more quantitative determination for it should be the focus of our continuing effort in the future.

Acknowledgment. The authors would like to thank the Chinese National Science Foundation for its valuable support of this work under award 61473220 and all the anonymous reviewers for their valuable comments.

References

1. Hawkins, D.M.: Identification of Outliers, Monographs on Applied Probability and Statistics. Chapman and Hall, London (1980)

2. Eskin, E., Arnold, A., Prerau, M., Portnoy, L., Stolfo, S.: A geometric framework for unsupervised anomaly detection: detecting intrusions in unlabeled data. In: *Data Mining for Security Applications* (2002)
3. Lane, T., Brodley, C.E.: Temporal sequence learning and data reduction for anomaly detection. *ACM Transactions on Information and System Security* **2**(3), 295–331 (1999)
4. Sheng, B., Li, Q., Mao, W., Jin, W.: Outlier detection in sensor networks. In: *Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 219–228 (2007)
5. Hodge, V.J., Austin, J.: A Survey of Outlier Detection Methodologies. *Artificial Intelligence Review* **22**, 85–126 (2004)
6. Chandola, V., Banerjee, A., Kumar, V.: Anomaly Detection: A Survey. *ACM Computing Surveys* **41**(3), Article 15 (2009)
7. Knorr, E.M., Ng, R.T.: Algorithms for mining distance-based outliers in large datasets. In: *Proceedings of the 24th VLDB Conference*, New York, USA, pp. 392–403 (1998)
8. Breuning, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: LOF: identifying density-based local outliers. In: *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, pp. 93–104 (2000)
9. Jiang, M.F., Tseng, S.S., Su, C.M.: Two-Phase Clustering Process for Outliers Detection. *Pattern Recognition Letters* **22**, 691–700 (2001)
10. Malik, J., Belongie, S., Leung, T., et al.: Contour and texture analysis for image segmentation. *International Journal of Computer Vision* **43**(1), 7–27 (2001)
11. Bach, F.R., Jordan, M.I.: Blind one-microphone speech separation: a spectral learning approach. In: *Proceedings of NIPS 2004*, Vancouver, BC, pp. 65–72 (2004)
12. Weiss, Y.: Segmentation using eigenvectors: a unified view. In: *International Conference on Computer Vision*, Corfu, pp. 975–982 (1999)
13. Ding, C., He, X., Zha, H., et al.: A min-max cut algorithm for graph partitioning and data clustering. In: *Proceedings of International Conference on Data Mining*, California, pp. 107–114 (2001)
14. Shi, J., Malik, J.: Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(8), 888–905 (2000)
15. Stoer, M., Wagner, F.: A simple min-cut algorithm. *Journal of the ACM* **44**(4), 585–591 (1997)
16. Hagen, L., Kahng, A.: New spectral methods for ratio cut partitioning and clustering. *IEEE Trans. Computer-Aided Design* **11**(9), 1074–1085 (1992)
17. Zelnik-Manor, L., Perona, P.: Self-tuning spectral clustering. In: *Proceedings of NIPS 2004*, Vancouver, BC, pp. 1601–1608 (2004)
18. Knorr, E.M., Ng, R.T., Tucakov, V.: Distance-based outliers: algorithms and applications. *VLDB Journal: Very Large Databases* **8**(3–4), 237–253 (2000)
19. Ramaswamy, S., Rastogi, R., Shim, K.: Efficient algorithms for mining outliers from large data sets. In: *Proceedings of the ACM SIGMOD Conference*, pp. 427–438 (2000)
20. Angiulli, F., Pizzuti, C.: Fast outlier detection in high dimensional spaces. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) *PKDD 2002*. LNCS (LNAI), vol. 2431, pp. 15–26. Springer, Heidelberg (2002)
21. UCI: The UCI KDD Archive. University of California, Irvine, CA. <http://kdd.ics.uci.edu/>
22. Aggarwal, C., Yu, P.: Outlier detection for high-dimensional data. In: *Proceedings of SIGMOD 2001*, Santa Barbara, CA, USA, pp. 37–46 (2001)

Machine Learning and Data Mining in Pattern
Recognition

12th International Conference, MLDM 2016, New York,
NY, USA, July 16-21, 2016, Proceedings

Perner, P. (Ed.)

2016, XIII, 807 p. 291 illus., Softcover

ISBN: 978-3-319-41919-0