Int. J. Mach. Learn. & Cyber. (2014) 5:469–477
DOI 10.1007/s13042-013-0202-4

ORIGINAL ARTICLE

A simple and effective outlier detection algorithm for categorical data

Xingwang Zhao · Jiye Liang · Fuyuan Cao

Received: 5 June 2013 / Accepted: 12 September 2013 / Published online: 27 September 2013

Abstract Outlier detection is an important data mining task that has attracted substantial attention within diverse research communities and the areas of application. By now, many techniques have been developed to detect outliers. However, most existing research focus on numerical data. And they can not directly apply to categorical data because of the difficulty of defining a meaningful similarity measure for categorical data. In this paper, a weighted density definition is given firstly, which takes account of the density and uncertainty of objects in every attributes simultaneously. Furthermore, a simple and effective outlier detection algorithm for categorical data based on the given weighted density is proposed. The corresponding time complexity of the algorithm is analyzed as well. Experimental results on real and synthetic data sets demonstrate the effectiveness and efficiency of our proposed algorithm.

Keywords Outlier detection · Categorical data · Weighted density · Information entropy

X. Zhao (✉) · J. Liang · F. Cao
Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan 030006, Shanxi, China
e-mail: zhaoxw84@163.com

J. Liang
e-mail: ljy@sxu.edu.cn

F. Cao
e-mail: cfy@sxu.edu.cn

1 Introduction

Different from traditional data mining task that attempts to find regular or frequent patterns, outlier detection targets to detect the rare data whose behavior are very exceptional when compared with the rest large amount of data [1]. One of the most widely accepted definitions of an outlier pattern is provided by Hawkins [2]: "An outlier is an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism".

Outlier detection has attracted a great deal of attention within diverse research communities and the areas of application. The importance of outlier detection is due to the fact that outliers can be indicative of bad data or malicious behavior in a wide variety of application domains. For example, abnormal behavior patterns in a computer network could mean that a hacked computer is sending out sensitive data to an unauthorized destination. Similarly, outliers in credit card transaction data could indicate credit card theft or misuse. Efficient detection of such outliers could reduce the risk of making poor decisions based on erroneous data, and aids in identifying and preventing the effects of malicious or faulty behavior [3]. Additionally, many data mining algorithms and statistical analysis techniques may not work well in the presence of outliers. Outliers may introduce skew or complexity into models of the data, which make it difficult to fit an accurate model to the data. Therefore, accurate and efficient removal of outliers may greatly enhance the performance of statistical techniques and data mining algorithms [4]. As can be seen, different domains have different reasons to detect outliers. They may be noise that we want to remove, since they obscure the true patterns we wish to discover, or they may be the very things in the data that we wish to discover. That is to say, "One person's noise is another

person's signal" [7]. Outlier detection has been a widely researched problem and finds immense use in a wide variety of application domains such as credit card, insurance, tax fraud detection in financial system, intrusion detection in computer networks, fault detection in safety critical systems, military surveillance for enemy activities and many other areas.

Over the years, a variety of outlier detection techniques have been developed in several research communities. The existing outlier detection techniques can be categorized as follows [5]. *Statistical distribution-based* methods [6] assume a distribution or probability model for the given data set (e.g., a normal or Poisson distribution) and then identify outliers with respect to the model using a discordancy test. Such methods do not work well in even moderately multivariate spaces, and the distribution of the measurement data is unknown in practice. In order to overcome the limitations imposed by these methods, Knorr et al [7–9] firstly introduced the notion of *distance-based* outliers. An object in a given data set is a distance-based outlier if at least a user-defined fraction of the points are further away than some user-defined minimum distance from that point. In other words, rather than relying on statistical tests, we can think of distance-based outliers as those objects that do not have "enough" neighbors, where neighbors are defined based on distance from the given object. The execution time of most distance-based outlier detecting methods is at least quadratic with respect to the number of objects, which may be unacceptable if the data set is very large or dynamic. Statistical and distance-based outlier detection methods both depend on the overall or "global" distribution of the given data set. However, data are usually not uniformly distributed. These methods encounter difficulties when analyzing data with rather different density distributions. This forms the basis of density-based local outlier detection. *Density-based* methods estimate the density distribution of the data and identify outliers as those lying in low-density regions. Another key idea of this approach to outlier detection is that, it assesses the degree to which an object is an outlier instead of a binary property. This degree of "outlierness" is computed as the local outlier factor (LOF) of an object. It is local in the sense that the degree depends on how isolated the object is with respect to the surrounding neighborhood. The disadvantage of this method is that it is very sensitive to parameters defining the neighborhood. *Deviation-based* outlier detection does not use statistical tests or distance-based measures to identify exceptional objects. Instead, it identifies outliers by examining the main characteristics of objects in a group. The objects that "deviate" from this description are considered as outliers. There are also *clustering-based* techniques used to detect the outliers [10]. The outliers detected by the clustering methods are considered as by-products and not optimized for outlier detection. However, most of the aforementioned techniques are geared towards data sets that are comprised of numerical attributes. These approaches cannot easily extend to categorical data because there is little sense in calculating distance among categorical data.

To tackle the above problem of detecting outliers in categorical data, several techniques have been recently developed in the literature [11–18]. Li et al. [11] proposed a distance-based outlier detecting algorithm using a new common neighbor-based distance to measure the proximity between categorical objects. The proposed algorithm consists of two steps, the neighbor-set generating step and the outlier mining step. The neighbor-set of the k nearest neighbors with similarity threshold θ to all objects is computed in the neighbor-set generation step. Both k and θ are two user-defined parameters. In the second step, an outlier factor of each object is computed by summing distance from its neighbors. The p objects with the largest values are returned to the user as outliers. This approach has two input prerequisite parameters, which are difficult to set in advance. Now, the concept of frequent items from association-rule mining have been used in outlier detection. Such methods consider the frequent or infrequent items of the data set. For example, He et al. [12] observe that, outliers are likely to be the objects that contain less frequent patterns in their itemsets. The procedure of the proposed algorithm includes an initial computation of the set of frequent patterns, using a pre-defined minimum support rate. For each object, all support rates of associated frequent patterns are summed up as the outlier factor of this object. The objects with lower factors are likely to be outliers. Contrary to the algorithm in [12], the algorithm proposed by Otey et al.[13] begins by collecting the infrequent items from the dataset. Based on the infrequent items, the outlier factors of the objects are computed. The objects with the largest scores are treated as outliers. The time complexity of both algorithms is determined by the frequent-item or infrequent-item generating processes, which is exponential to the number of categorical attributes. Recently, information theory has been used to detect outliers. The algorithms in [14, 15] employ information entropy to measure the disorder of a dataset after removing the outliers, and define the problem of outlier detection as an optimization problem. To address this problem, a local-search heuristic-based algorithm and a greedy algorithm are applied to minimize the objective function, respectively. In recent years, outlier detection techniques for categorical data have also been developed in rough set research communities. Jiang et al. [16] proposed a RMF (rough membership function)-based outliers detection algorithm. The outlier detection is based on approximation theory where outliers are detected in the boundary region.

When given a set of indiscernibility relations on the objects U, if the values of rough membership function of x with respect to X (a subset of the objects U) under these indiscernibility relations are always small, then the object x may be considered as an outlier. In 2009, Jiang et al. [17] proposed a sequence-based outliers detection algorithm. As is well known, if attribute subset B decreases gradually, then the granularity of partition on the objects U will become coarser, and for every object $x \in U$ the corresponding equivalence class of x will become bigger. So when there is an object in U whose equivalence class always does not vary or only increases a little in comparison with the other objects in U, then this object can be considered as a sequence-based outlier. However, recent research proved that the existing methods are not suitable for data mining applications in practice. Because many existing methods suffer from low effectiveness and low efficiency due to high dimensionality and large size of the dataset. And several user-defined parameters are also often required to define. The parameter-laden results are heavily dependent on suitable parameter settings, which are very difficult to estimate without background knowledge about the data. Therefore, it still remains a challenge to propose an effective outlier detection algorithm for categorical data.

In this paper, by extending the average density given in our previous work [19], we give a weighted density definition for categorical data based on information entropy. The given weighted density takes account of the density and uncertainty of objects in every attributes simultaneously. Furthermore, a simple and effective outlier detection algorithm for categorical data based on the given weighted density is proposed. The corresponding time complexity of the algorithm is also analyzed. In the experimental section, we compare the proposed algorithm with existing outlier detection methods with respect to outlier detection accuracy and time consumption. Experimental results illustrate the superiority of our proposed algorithm. It can be applied to large data sets with high dimensions for its linear time complexity with respect to the number of data objects and dimensions.

The remainder of this paper is organized as follows. In Sect. 2, we present some preliminaries used throughout the paper. In Sect. 3, a weighted density-based outlier detection algorithm for categorical data is proposed. Experimental results are shown in Sect. 4. Finally, Sect. 5 provides a conclusion and future work.

2 Preliminaries

In this section, several basic concepts are reviewed. In the real world, a large portion or the entirety of the data sets is often presented in terms of categorical attributes. Examples of such data sets include transaction data, financial records in commercial banks, demographic data, etc. In general, categorical data are assumed to be stored in a table, where each row (tuple) represents facts about an object. More formally, a categorical data table is described by a quadruple $DT = (U, A, V, f)$, where:

(1) U is a nonempty set of objects, called a universe;
(2) A is a nonempty set of attributes;
(3) V is the union of attribute domains, i.e., $V = \bigcup_{a \in A} V_a$, where V_a is the value domain of attribute a and it is finite and unordered, e.g., for any $p, q \in V_a$, either $p = q$ or $p \neq q$;
(4) $f: U \times A \to V$ is an information function such that, for any $a \in A$ and $x \in U$, $f(x, a) \in V_a$.

Definition 1 Let $DT = (U, A, V, f)$ be a categorical data table and $P \subseteq A$. A binary relation $IND(P)$, called indiscernibility relation, is defined as

$$IND(P) = \{(x, y) \in U \times U | \forall a \in P, f(x, a) = f(y, a)\}. \quad (1)$$

Two objects are indiscernible in the context of a set of attributes if they have the same values for those attributes. $IND(P)$ is an equivalence relation on U and $IND(P) = \cap_{a \in P} IND(\{a\})$.

The relation $IND(P)$ induces a partition of U, denoted by $U/IND(P) = \{[x]_P | x \in U\}$, where $[x]_P$ denotes the equivalence class determined by x with respect to P, i.e., $[x]_P = \{y \in U | (x, y) \in IND(P)\}$.

3 A weighted density-based outlier detection algorithm

In this section, after reviewing the average density and demonstrating the need for attribute weighting, we will give a weighted density definition of the object by incorporating the uncertainty measure. Furthermore, a simple and effective outlier detection algorithm for categorical data is designed.

For categorical data, the outliers are intuitively those points with highly irregular or infrequent values. The higher the infrequency of the object value on each attribute, the more likely the object is an outlier. Additionally, an "ideal" outlier in a categorical data set is one whose each and every attribute value is extremely irregular or infrequent. The infrequent-ness can be measured by computing the average density of the object based on the equivalence class. The average density of an object in a given data set is defined as follows [19].

Definition 2 Let $DT = (U, A, V, f)$ be a categorical data table. For any object $x \in U$, the average density of x in U with respect to A is defined as

$$ADens(x) = \frac{\sum_{a \in A} ADens_a(x)}{|A|}, \quad (2)$$

where $ADens_a(x)$ is the density of object x in U with respect to the attribute a, given by

$$ADens_a(x) = \frac{|[x]_{\{a\}}|}{|U|}. \quad (3)$$

and $|A|$ and $|U|$ represent the number of attributes and objects, respectively.

In the universe, the less $ADens(x)$ is, if can be expressed in a graph, the less the number of objects around the object x is. In other words, a less $ADens(x)$ value implies that the object x is more possible to be an outlier. In the above definition, the weights to all the attributes are equal. However, the effect of different attributes on causing the outlier degree of an object is not equal. In some extreme case, only a few attributes can decide an object to be an outlier. That is to say, different attributes often contribute differently to form the overall structure of the data in real applications. It is well known that the expression of data distribution is usually uncertain. And the uncertainties come from disorder, vagueness, approximate expression, and so on. One of the most common uncertainty measures of data sets is information entropy or its variants [20, 21]. The entropy of a system defined by Shannon gives a measure of uncertainty about its actual structure [22]. Since the aim of outlier detection is to detect the rare objects who behave in an unexpected way or have abnormal properties. And uncertainty can be considered as a kind of abnormal property. Therefore, the information entropy can be used for outlier detection.

In the categorical domain, Liang et al. [23] introduced a new information entropy called complementary entropy and used it to measure information content and uncertainty for a categorical data table. Unlike the logarithmic behavior of Shannon's entropy, the complement entropy not only can measure the uncertainty, but also the fuzziness. Recently, it has been used in a variety of applications for categorical data, including clustering analysis [24, 25], feature selection [26, 27], rule evaluation [28], uncertainty measure [29], etc. The complement entropy for categorical data is defined as follows.

Definition 3 Let $DT = (U, A, V, f)$ be a categorical data table, $P \subseteq A$ and $U/IND(P) = \{X_1, X_2, ..., X_m\}$. The complement entropy with respect to P is defined as

$$E(P) = \sum_{i=1}^{m} \frac{|X_i|}{|U|}\frac{|X_i^c|}{|U|} = \sum_{i=1}^{m} \frac{|X_i|}{|U|}\left(1 - \frac{|X_i|}{|U|}\right), \quad (4)$$

where X_i^c denotes the complement set of X_i, i.e., $X_i^c = U - X_i$; $\frac{|X_i|}{|U|}$ represents the probability of X_i within the universe U; $\frac{|X_i^c|}{|U|}$ is the probability of the complement set of X_i within the universe U and $|U|$ represents the number of objects.

According to Definition 3, given a categorical data table $DT = (U, A, V, f)$, for $a \in A$, the higher the value of $E(\{a\})$, the more out-of-order the distribution of V_a. The complement entropy $E(\{a\})$ is maximal when the V_a has a uniform distribution, which means that it possesses maximal uncertainty on the attribute a. Based on the above idea, a weighted density definition is given as follows.

Definition 4 Let $DT = (U, A, V, f)$ be a categorical data table. For any object $x \in U$, the weighted density of x in U with respect to A is defined as

$$WDens(x) = \sum_{a \in A} ADens_a(x) \cdot W(\{a\}), \quad (5)$$

where $W(\{a\})$ is a weighting function with respect to attribute $a \in A$, given by

$$W(\{a\}) = \frac{1 - E(\{a\})}{\sum_{l \in A}(1 - E(\{l\}).} \quad (6)$$

The weighting function W in the above definition is designed to measure the distribution of the value domain on each individual attribute by utilizing information entropy. Suppose that there is a value domain of attribute a whose distribution is uniformly. The V_a that contains the maximum uncertainty provides more outlier characteristics. Then we should give more importance to the attribute a. Therefore, this attribute should be assign a smaller weight in weighted density. In the above definition, the weight of every attribute has been normalized from zero to one, and the sum of all weight is one. Furthermore, it is easy to verify that the range of the weighted density value is [0, 1], because the range of both the density of object and the weighting function is [0, 1].

Following the ideas given above, the weighted density can be used as a good indicator to determine whether an object is an outlier or not. Furthermore, the smaller the weighted density value of an object x, the more the likelihood of x to be an outlier.

Definition 5 (Weighted density-based outliers) Let $DT = (U, A, V, f)$ be a categorical data table, and θ be a given threshold value. For any object $x \in U$, if $WDens(x) < \theta$, then the object x is called a weighted density-based outlier.

In Definition 5, the outlier threshold θ is important in the process of outliers detecting. However, it is hard to define a uniform value that is applied on all datasets. Here are some hints to provide the clues to set this parameter: (1) detecting outliers in stable data sets requires a high

Table 1 A weighted density-based outlier detection algorithm for categorical data

	Input: A categorical data table $DT = (U, A, V, f)$ and threshold value θ. **Output:** A set O of weighted density-based outliers.
1	Let $O = \varnothing$
2	For every $a \in A$
3	{
4	Compute the partition $U/IND(\{a\})$ according to
5	definition 1;
6	Compute the complement entropy $E(\{a\})$ according
7	to definition 3;
8	}
9	For every $x \in U$
10	{
11	For every $a \in A$
12	{
13	Compute the average density $ADens_a$ according
14	to definition 2;
15	}
16	Compute the weighted density$WDens(x)$ according
17	to definition 4;
18	If $WDens(x) < \theta$, then $O = O \bigcup x$.
19	}
20	Return O.

Table 2 A categorical data set

U\A	a	b	c
x_1	A	E	M
x_2	A	D	N
x_3	B	G	M
x_4	C	D	N
x_5	C	G	M
x_6	C	F	N

threshold value; (2) detecting outliers in unstable data sets requires a low threshold value. Therefore, if we can obtain prior knowledge of the data from domain experts, the prior knowledge can help us to set proper parameter values.

Based on the above mentioned formulations and notations, a weighted density-based outlier detection algorithm for categorical data (abbreviated as WDOD) is described in Table 1.

The following is the time complexities of the WDOD algorithm. In the WDOD algorithm, we use a method given in [30] for partition with time complexity being $O(|U||A|)$. So, the time complexity of computing complement entropy, i.e., attributes weighting, is $O(|U||A|)$. Therefore, the overall time complexity of the proposed algorithm is $O(|U||A|)$, which is linearly scalable to the number of objects and attributes.

In the following, the above definitions and the process of detecting outliers in a dataset are illustrated in Example 1.

Example 1 Consider the data in Table 2. This is a categorical data table, where $U = \{x_1, x_2, ..., x_6\}$ and $A = \{a, b, c\}$.

According to Definition 1, the partitions of U with respect to different attributes are given by $U/IND(\{a\}) = \{\{x_1, x_2\}, \{x_3\}, \{x_4, x_5, x_6\}\}$, $U/IND(\{b\}) = \{\{x_1\}, \{x_2, x_4\}, \{x_3, x_5\}, \{x_6\}\}$ and $U/IND(\{c\}) = \{\{x_1, x_3, x_5\}, \{x_2, x_4, x_6\}\}$. By Definition 3, one have that

$$E(\{a\}) = \frac{2}{6}\left(1 - \frac{2}{6}\right) + \frac{1}{6}\left(1 - \frac{1}{6}\right) + \frac{3}{6}\left(1 - \frac{3}{6}\right) = \frac{11}{18},$$

$$E(\{b\}) = 2 \times \frac{1}{6}\left(1 - \frac{1}{6}\right) + 2 \times \frac{2}{6}\left(1 - \frac{2}{6}\right) = \frac{13}{18}$$

and

$$E(\{c\}) = \frac{3}{6}\left(1 - \frac{3}{6}\right) + \frac{3}{6}\left(1 - \frac{3}{6}\right) = \frac{9}{18}.$$

Obviously, $E(\{c\}) < E(\{a\}) < E(\{b\})$. In other words, $E(\{b\})$ achieves its maximal value, which means that the attribute b contains maximal uncertainty and provides more outlier characteristics. Therefore, this attribute should contribute more in the process of outliers detecting. By Eq. (6), the weights of every attribute are given by $W(\{a\}) = \frac{7}{21}, W(\{b\}) = \frac{5}{21}$ and $W(\{c\}) = \frac{9}{21}$. According to Definition 4, we can have the following weighted density values of objects in U:

$$WDens(x_1) = \frac{2}{6} \times \frac{7}{21} + \frac{1}{6} \times \frac{5}{21} + \frac{3}{6} \times \frac{9}{21} = 0.3651,$$
$$WDens(x_2) = \frac{2}{6} \times \frac{7}{21} + \frac{2}{6} \times \frac{5}{21} + \frac{3}{6} \times \frac{9}{21} = 0.4048,$$
$$WDens(x_3) = 0.3492, WDens(x_4) = 0.4603,$$
$$WDens(x_5) = 0.4603, WDens(x_6) = 0.4206.$$

If the outlier threshold value θ is set 0.4. Then, the objects x_1 and x_3 are considered as outliers.

4 Experimental results

In this section, we conduct effectiveness and efficiency tests to analyze the performance of the proposed algorithm. All the experiments are conducted on a PC with an Intel Pentium 4–2.66 GHz core 2 Quad CPU and 4 G memory running the Windows XP SP3 operating system. The proposed algorithm and the comparative algorithms are coded in Matlab 7.0 programming language.

Since most of the existing outlier detection methods for categorical data need some user-defined parameters in advance. The parameter-laden results are heavily dependent on suitable parameter settings, which are very difficult to set without background knowledge about the data. For reasons of fairness, the proposed algorithm is compared with the local-serach heuristic algorithm (denoted as LSA) [14] and the sequence-based outlier detection algorithm (denoted as SEQ) [17], which only need one parameter, i.e., the number of outliers. In Sect. 4.1, the real data sets downloaded from the UCI machine learning repository [31] are described, and the results of effectiveness tests on these data sets are reported. For the efficiency test, we conduct evaluations on synthetic data sets to show how running time increases with the number of objects, the number of attributes and the number of outliers in Sect. 4.2.

4.1 Effectiveness analysis

In order to test the effectiveness of an outlier detection method, Aggarwal and Yu [32] proposed a practicable way to test how well the method worked. That is, we can run the outlier detection method on a given data set and test the percentage of points which belonged to one of the rare classes. Since the class labels of each object in the test data sets are known in advance, the points belonged to the rare classes are considered as outliers. If the method works well, we expect that such abnormal classes would be over-represented in the set of points found.

In this subsection, following the experimental setup mentioned above, we use four real life data sets to demonstrate the performance of the proposed algorithm (WDOD) against the existing algorithms, i.e., LSA [14] and SEQ [17]. The data sets and the corresponding results are described as follows, respectively.

Lymphography This data set contains 148 objects and 18 categorical attributes and class attribute. These objects are partitioned into four classes, i.e.,"normal find" (1.35 %), "metastases" (54.73 %), "malign lymph" (41.22 %) and "fibrosis" (2.7 %). These rare objects in classes 1 and 4 ("normal find" and "fibrosis") are considered as the outliers.

The experimental results produced by the WDOD algorithm against the LSA algorithm and SEQ algorithm on this data set are summarized in Table 3. Here, the top ratio is ratio of the number of objects specified as top-k outliers to that of the objects in the dataset. The coverage is ratio of the number of detected rare classes to that of the rare classes in the data set. For example, we let the WDOD algorithm find the top six outliers with the top ratio of 4 %. By examining these six objects, we found that 5 of them belonged to the rare classes. In contrast, when we ran the SEQ algorithm on this data set, we found that only 4 of top 6 outliers belonged to rare classes.

Table 3 Experimental results on lymphography data set

Top ratio (number of objects)	Number of rare classes included (coverage)		
	LSA	SEQ	WDOD
3 % (4)	4 (67 %)	4 (67 %)	4 (67 %)
4 % (6)	5 (83 %)	4 (67 %)	5 (83 %)
5 % (8)	6 (100 %)	5 (83 %)	5 (83 %)
8 % (12)	6 (100 %)	5 (83 %)	6 (100 %)
10 % (15)	6 (100 %)	6 (100 %)	6 (100 %)

Table 4 Experimental results on wisconsin breast cancer data set

Top ratio (number of objects)	Number of rare classes included (coverage)		
	LSA	SEQ	WDOD
1 % (4)	4 (10.3 %)	3 (7.7 %)	3 (7.7 %)
2 % (8)	8 (20.5 %)	7 (17.8 %)	7 (17.8 %)
4 % (16)	14 (35.9 %)	14 (35.9 %)	14 (35.9 %)
6 % (24)	21 (53.8 %)	19 (48.7 %)	18 (46.2 %)
8 % (32)	28 (71.8 %)	23 (59.1 %)	24 (61.5 %)
10 % (40)	30 (76.9 %)	28 (71.8 %)	30 (76.9 %)
12 % (48)	35 (89.7 %)	33 (84.6 %)	34 (87.2 %)
14 % (56)	39 (100 %)	37 (94.9 %)	39 (100 %)
16 % (64)	39 (100 %)	38 (97.4 %)	39 (100 %)
18 % (72)	39 (100 %)	39 (100 %)	39 (100 %)

From Table 3, the performance of the WDOD algorithm outperformed that of the SEQ algorithm, and a little weaker than LSA.

Wisconsin breast cancer This data set was collected by Dr. William H. Wolberg at the University of Wisconsin Madison Hospitals. There are 699 records in this data set. Each record has nine attributes, which are graded on an interval scale from a normal state of 1−10, with 10 being the most abnormal state. In this database, 241 records are malignant and 458 records are benign. According to the experimental technique of Harkins et al. [33], we randomly remove some of the records to form a very unbalanced distribution. The resultant data set had 39 (8%) malignant objects and 444 (92%) benign objects. That is to say, there are 39 outliers in this data set. The experimental results are summarized in Table 4.

From Table 4, we can see that for the wisconsin breast cancer data set, the performance of the WDOD algorithm is better than SEQ, and very close to LSA that has large computation time. For example, the proposed WDOD algorithm and LSA detect 39 true outliers from 56 expected

outliers. While the SEQ algorithm only detect 37 true outliers.

Letter recognition This data set contains character image features of 26 capital letters in the English alphabet. Each object is described by 16 attributes which are integer valued and seen as categorical attributes in the experiment. In order to form an imbalanced data for outlier detection, we choose all the objects labeled "A" and some of objects labeled "B" to form a new data. In the new data set, there are 839 objects in all, including 789 objects with label "A" and 50 objects with label "B". And the objects with label "B" are considered as outliers. The experimental results are summarized in Table 5.

Mushroom This data set includes descriptions of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family. Each species is identified as definitely edible, definitely poisonous. This data set contains 8,124 samples, which are classified into two classes. One class is edible with 4,208 (51.8 %) samples, and another class is poisonous with 3,916 (48.2 %) samples. All 22 attributes are nominally valued. To make the data set more imbalanced, we added 76 outliers to the original data to form a new data set. The final data set contains 8,200 total objects, and 76 outliers or 0.93 % of new data set. The experimental results produced by the WDOD algorithm against the LSA algorithm and SEQ algorithm on this data set are summarized in Table 6.

From Tables 5, 6, we can see that the performance of the WDOD algorithm is better than SEQ and LSA.

Table 5 Experimental results on letter recognition data set

Top ratio (number of objects)	Number of rare classes included (coverage)		
	LSA	SEQ	WDOD
3.6 % (30)	12 (24 %)	21 (42 %)	23 (46 %)
6 % (50)	13 (26 %)	27 (54 %)	28 (56 %)
8.3 % (70)	16 (32 %)	38 (76 %)	37 (74 %)
10.7 % (90)	31 (62 %)	41 (82 %)	42 (84 %)
13.1 % (110)	36 (72 %)	45 (90 %)	46 (92 %)
15.5 % (130)	42 (84 %)	47 (94 %)	48 (96 %)
17.9 % (150)	47 (94 %)	48 (96 %)	49 (98 %)
20.3 % (170)	48 (96 %)	49 (98 %)	50 (100 %)
22.6 % (190)	49 (98 %)	50 (100 %)	50 (100 %)
23.8 % (200)	50 (100 %)	50 (100 %)	50 (100 %)

Table 6 Experimental results on mushroom data set

Top ratio (number of objects)	Number of rare classes included (coverage)		
	LSA	SEQ	WDOD
0.37 % (30)	26 (34.2 %)	30 (39.5 %)	30 (39.5 %)
0.64 % (53)	39 (51.3 %)	42 (55.3 %)	53 (69.8 %)
0.93 % (76)	50 (65.8 %)	56 (73.7 %)	68 (89.5 %)
1.1 % (87)	57 (75 %)	62 (81.6 %)	74 (97.4 %)
1.2 % (99)	65 (85.5 %)	72 (94.7 %)	75 (98.7 %)
1.3 % (110)	72 (94.7 %)	75 (98.7 %)	76 (100 %)
1.5 % (122)	73 (96.1 %)	75 (98.7 %)	76 (100 %)
1.6 % (130)	76 (100 %)	76 (100 %)	76 (100 %)

4.2 Efficiency analysis

Experiment results on time consumption of the three outlier detection algorithms with increasing number of objects, attributes and outliers are reported in this subsection. For the test of efficiency, we employ a synthetic data generator [34] to generate a few categorical data sets with different number of data points and attributes. In all synthetic data sets, each attribute possesses five different values. The number of data points varies from 1,000 to 10,000, and the dimensionality is in the range of 10–50.

We conduct three types of test to see the change of each algorithm's performance as parameters change, e.g., the size of the data set, the data set dimensionality and the number of outliers. Figure 1 shows the scalability of the three algorithms with data size. This study fixes the dimensionality to 10, and the number of outliers to 20, and also varies the data size from 1,000 to 10,000. It can be seen that all the three outlier detection methods are approximate linear with respect to the data size. However, the increasing rate of the execution time on the WDOD is much slower than LSA and SEQ algorithms. Therefore, the proposed WDOD algorithm can ensure efficient execution when the data size is large.

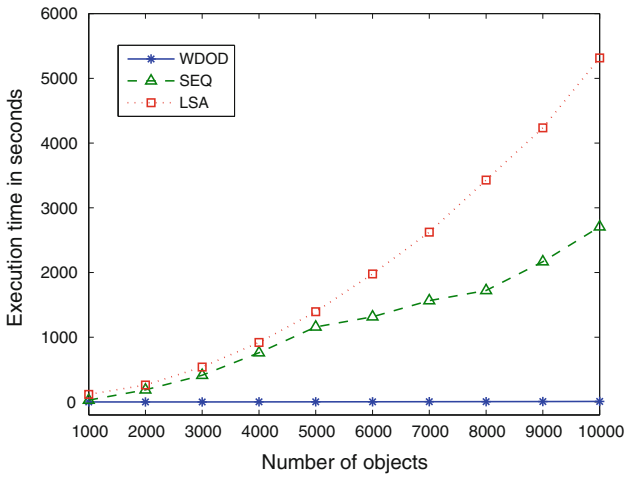

Fig. 1 Execution time comparison with the increasing number of objects

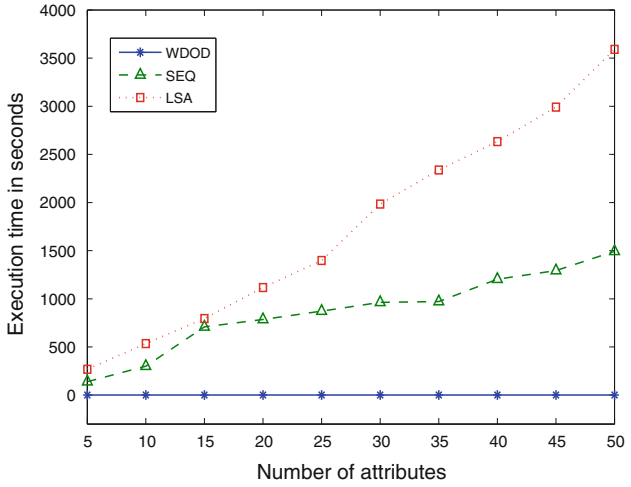

Fig. 2 Execution time comparison with the increasing number of attributes

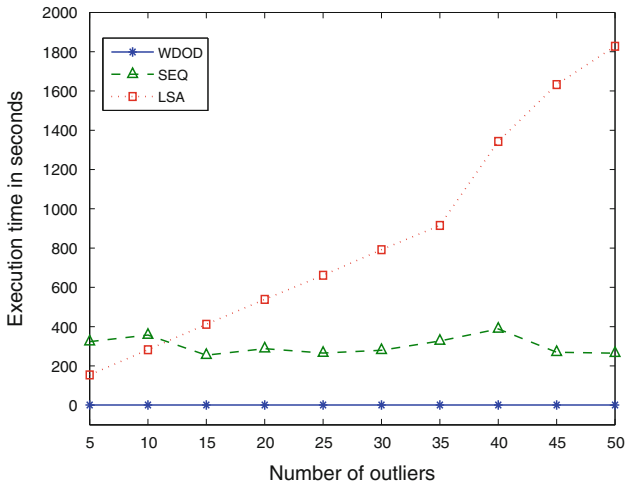

Fig. 3 Execution time comparison with the increasing number of outliers

Figure 2 shows the scalability with data dimensionality of the three algorithms. We fix the data size to 3,000, and the number of outliers to 20, and also vary the number of dimensions from 5 to 50. Similar to Fig. 1, all the three outlier detection methods are also approximate linear with respect to the data dimensionality. However, the increasing rate of the execution time on the WDOD is much slower than LSA and SEQ algorithms.

Figure 3 shows the scalability with the number of outliers of the three algorithms. In this study, the data size is fixed to 3,000, and the dimensionality is fixed to 10, and the number of outliers varies from 5 to 50. According to the figure, the execution time of the WDOD and SEQ algorithms does not increase when the number of outliers increases. In contrast, the execution time of the LSA method increases when the number of outliers increases from 10 to 50. This is owing to the reason that the LSA method need to scan the data set k (the number of outliers) times to detect k outliers.

As we see from these experiment results with real and artificially generated data, the WDOD algorithm outperforms the SEQ algorithm and approximates very well of LSA in terms of outlier detection accuracy. Moreover, the WDOD algorithm requires a short time with respect to the data size, data dimensionality, and the target number of outliers, in comparison to the other two algorithms. These experiment tests suggest that the WDOD algorithm is particularly appropriate for large data set with high dimensionality, and also suitable for data sets with high percentage of outliers.

5 Conclusion and future work

Outlier detection is becoming critically important in many research communities and application domains. In this paper, a weighted density for measuring the uncertainty of every attributes and the density of each object was presented. Furthermore, we gave a simple and effective algorithm for outlier detection in categorical data. Experiment results on real data sets and large synthetic data sets show that the proposed algorithm is superior to existing algorithms in detecting outliers for categorical data, especially in terms of efficiency. Extending our work for dynamic and mixed data is the focus of our future work.

Acknowledgements The authors are very grateful to the anonymous reviewers and editor. Their many helpful and constructive comments and suggestions helped us significantly improve this work. This work was supported by the National Natural Science Foundation of China (No. 71031006), the Foundation of Doctoral Program Research of Ministry of Education of China (No. 20101401110002), the Construction Project of the Science and Technology Basic Condition Platform of Shanxi Province (No. 2012091002-0101) and Shanxi Scholarship Council of China (No. 2013-101).

References

1. Chandola V, Banerjee A, Kumar V (2009) Anomaly detection: A survey. ACM Comput Surv 41(3):Article 15
2. Hawkins D (1980) Identification of outliers. Chapman and Hall, London
3. Kumar V (2005) Parallel and distributed computing for cybersecurity. IEEE Distrib Syst Online 6(10). doi:10.1109/MDSO.2005.53
4. Gamberger D, Boskovic R, Lavrac N, Groselj C (1999) Experiments with noise filtering in a medical domain. In: Proceedings of the 16th international conference on machine learning
5. Han JW, Kamber M (2011) Data mining concepts and techniques, 3rd edn. Morgan Kaufmann Publishers Inc, San Francisco
6. Barnett V, Lewis T (1994) Outliers in statistical data. John Wiley, Chichester

7. Knorr E, Ng RT (1998) Algorithms for mining distance-based outliers in large datasets. In: Proceedings of the 24th VLDB conference, New York, pp 392–403
8. Knorr EM, Ng RT (1999) Finding intentional knowledge of distance-based outliers. In: Proceedings of 25th international conference on very large databases, Edinburgh, Scotland, pp 211–222
9. Knorr EM, Ng RT, Tucakovand V (2000) Distance-based outliers: algorithms and applications. VLDB J 8(3–4):237–253
10. Tang CL, Wang SG, Xu W (2010) New fuzzy c-means clustering model based on the data weighted approach. Data Knowl Eng 69:881–900
11. Li SX, Lee R, Lang SD (2007) Mining distance-based outliers from categorical data. In Proceedings of the 7th IEEE international conference on data mining workshops, Washington, pp 225–230
12. He ZY, Xu XF, Huang JZ, Deng SC (2005) FP-outlier: frequent pattern based outlier detection. Comput Sci Inf Syst 2(1):103–118
13. Otey ME, Ghoting A, Parthasarathy S (2006) Fast distributed outlier detection in mixed-attribute data sets. Data Min Knowl Discov 12:203–228
14. He ZY, Deng SC, Xu XF (2005) An optimization model for outlier detection in categorical data. In: Proceedings of the 2005 international conference on advances in intelligent computing, Hefei, pp 400–409
15. He ZY, Deng SC, Xu XF, Huang JZ (2006) A fast greedy algorithm for outlier mining. In: Proceedings of the 10th Pacific-Asia conference on knowledge and data discovery, pp 567–576
16. Jiang F, Sui YF, Cao CG (2008) A rough set approach to outlier detection. Int J Gen Syst 37(5):519–536
17. Jiang F, Sui YF, Cao CG (2009) Some issues about outlier detection in rough set theory. Expert Syst Appl 36(3):4680–4687
18. Jiang F, Sui YF, Cao CG (2010) An information entropy-based approach to outlier detection in rough sets. Expert Syst Appl 37(9):6338C6344
19. Cao FY, Liang JY, Bai L (2009) A new initialization method for categorical data clustering. Expert Syst Appl 36(7):10223–10228
20. Liang X, Wei CP (2013) An Atanassov's intuitionistic fuzzy multi-attribute group decision making method based on entropy and similarity measure. Int J Mach Learn Cybern. doi:10.1007/s13042-013-0178-0
21. Guan PP, Yan H (2012) A hierarchical multilevel thresholding method for edge information extraction using fuzzy entropy. Int J Mach Learn Cybern 3(4):297–305
22. Shannon CE (1948) A mathematical theory of communiction. Bell Syst Tech J 27(3–4):379–423
23. Liang JY, Chin KS, Dang CY (2002) A new method for measuring uncertainty and fuzziness in rough set theory. Int J Gen Syst 31(4):331–342
24. Liang JY, Zhao XW, Li DY, Cao FY, Dang CY (2012) Determining the number of clusters using information entropy for mixed data. Pattern Recognit 45(6):2251-2265
25. Cao FY, Liang JY, Li DY, Zhao XW (2013) A weighting k-modes algorithm for subspace clustering of categorical data. Neurocomputing 108:23–30
26. Qian YH, Liang JY, Pedrycz W, Dang CY (2010) Positive approximation: an accelerator for attribute reduction in rough set theory. Artif Intell 174(9-10):597–618
27. Liang JY, Wang F, Dang CY, Qian YH (2012) A group incremental approach to feature selection applying rough set technique. IEEE Trans Knowl Data Eng. doi:10.1109/TKDE.2012.146
28. Qian YH, Liang JY, Li DY, Zhang HY, Dang CY (2008) Measures for evaluating the decision performance of a decision table in rough set theory. Inf Sci 8(1):181–202
29. Liang JY, Shi ZZ, Li DY, Wierman MJ (2006) The information entropy, rough entropy and knowledge granulation in incomplete information system. Int J Gen Syst 35(6):641–654
30. Xu ZY, Liu ZP, Yang BR, Song W (2006) A quick attribute reduction algorithm with complexity of max($O(|C||U|)$, $O(|C|2|U/C|)$). Chin J Comput 29(3):391–398
31. UCI Machine Learning Repository 2012 http://archive.ics.uci.edu/ml/datasets.html
32. Aggarwal CC, Yu PS (2001) Outlier detection for high dimensional data. In: Proceedings of the 2001 ACM SIGMOD international conference on managment of data, California, pp 37–46
33. Hawkins S, He HX, Williams G, Baxter R (2002) Outlier detection using replicator neural networks. In: Proceedings of the 5th international conference and data warehousing and knowledge discovery
34. Cristofor D, Simovici D (2002) Finding median partitions using information-theoretical algorithms. J Univers Comput Sci 8(2):153–172 (software at http://www.cs.umb.edu/~dana/GAClust/index.html)