Automated Entropy Value Frequency (AEVF) Algorithm for Outlier Detection in Categorical Data

USMAN QAMAR
Computer Engineering Department
College of Electrical and Mechanical Engineering,
National University of Sciences and Technology (NUST)
Islamabad
PAKISTAN
usmanq@cheme.nust.edu.pk
http://www.nust.edu.pk

Abstract: - Outlier detection has been a very important concept in data mining. The aim of outlier detection is to find those objects that are of not the norm. There are many applications of outlier detection from network security to detecting credit fraud. However most of the outlier detection algorithms are focused towards numerical data and do not perform well when applied to categorical data. In this paper, we propose an automated outlier detection algorithm which specifically caters for categorical data.

Key-Words: - Outlier Detection, Entropy, Categorical Data, Numerical Data

1 Introduction

A definition of outliers is [1]: ‘an outlier is an observation/object that when compared to other observations/ojects are so different that it requires an analysis’.

Outliers generally have unusually large or small values when compared with others in a dataset. There may be different reasons for outliers varying from incorrect measurements, wrong data entry, or the most common being different from rest of the population. If the measurement is correct, it represents a rare occurrence. Examples of outliers include unusual credit card transactions, or patients who show abnormal symptoms.

Outlier detection has primarily focused on datasets which consists of numerical attributes. Outlier detection for categorical attributes such as sex, employment, marriage status is much more complicated. There exists no straightforward method which allows for the mapping of categorical attributes to numerical attributes. For example suppose for marriage status, we may have three categories, “married”, “widow” and “divorced”. Now if these are mapped to 1, 2, and 3 respectively then this does not make any sense for the purpose of outlier detection. However recent research has focused on categorical attributes (e.g. [3], [4] and [5]).

Another issue with outlier detection is predicing the optimal number of outliers that are to be generated. However, there is no way for the user to know this value apart from via hit and miss methods. As the optimal number of outliers varies from dataset to dataset, it is not possible to predict the number of outliers or to define a standard or fixed number of outliers that can be applicable for every dataset. Clearly an automated process is needed which will automatically generate the optimal number of outliers.

In this paper, we propose an outlier detection technique for categorical data which is based on entropy, called Automated Entropy Value Frequency (AEVF). AEVF is automated in that it requires no user input and will always generate the optimal number of outliers.

The paper is split into the following sections: In Section 2, overview of literature is given. In Section 3, we describe AEVF. Section 4 contains experimental work followed by analysis and results. Finally, conclusion is provided.

2 Literature Review

Information theory [6] based techniques analyze the information content of a dataset by applying...
Different information theory measures such as entropy. They are based on the concept that rare records or outliers will significantly alter the information content of the data because of their extremely different nature.

A framework to detect outliers using entropy has been proposed by Lee and Xiang [7]. Outlier detection in entropy is based on the observation that ‘removing outliers from a dataset will result in a dataset that is less dissimilar’ [7].

Entropy $E(X)$ can be defined as

$$E(X) = - \sum_{x \in S(X)} p(x) \log (p(x))$$

Where $X$ is a random variable, and $S(X)$ range of values that $X$ can have, and $p(x)$ the probability function of $X$.

By the definition of probability, [6] the following relation must hold:

$$\sum_{x} p(x) = 1$$

The entropy of a multivariable vector $x = \{X_1, \ldots, X_m\}$ can be computed as shown in Equation (2):

$$E(x) = - \sum_{x_1, \ldots, x_m} p(x_1, \ldots, x_m) \log (p(x_1, \ldots, x_m))$$

If the records are considered independent of each other, then Equation (2) can be rewritten as Equation (3). This implies that entropy can be computed as the sum of entropies of all the attributes:

$$E(x) = - \sum_{x_1, \ldots, x_m} p(x_1, \ldots, x_m) \log (p(x_1, \ldots, x_m))$$

$$= E(X_1) + E(X_2) + \ldots + E(X_m)$$

Working of LSA can be defined as following

- For a dataset $DS$, $k$ outliers are to be detected using entropy.
- The number of outliers to be generated ($k$) is user defined.
- The initial set of outliers (SO) is empty and all the dataset’s records are marked as non-outliers.
- $k$ scans are carried out to select $k$ records as outliers.
- During each scan, each record tagged as a non-outlier is temporarily removed from the dataset and the change in entropy is calculated. $T$
- The record that achieves the maximum decrease in entropy by removing that record, is selected as an outlier and added to SO.
- This continues for each scan until the size of OS reaches the defined value of $k$. Using an LSA has been shown to produce the best results in detecting outliers both in terms of accuracy and speed – when compared with other outlier detection techniques [7].

The obvious disadvantage of LSA is that it is dependent upon user input to specify the number of outliers. However, there is no way for the user to know this value apart from via hit and miss methods. As the optimal number of outliers varies from dataset to dataset, it is not possible to predict the number of outliers or to define a standard or fixed number of outliers that can be applicable for every dataset.

In order to overcome these limitations, an extended version of the LSA algorithm is proposed and then implemented in this paper.

3 Automated Entropy Value Frequency (AEVF)

This extended version of the algorithm introduces the new terms entropy difference gap and max entropy gap. The entropy difference gap is the difference in the values of change of entropy between one record and the next, while the max entropy gap is a user input. The max entropy gap is the maximum entropy difference gap that can exist as defined by the user. If the entropy difference gap becomes larger than the max entropy gap, the algorithm terminates.
The algorithm is a two-step process:

1. Generating entropy change values. In the first step, the change in entropy value for each record in a dataset is generated and stored in a table. Once all records have been processed, the table is updated so that the record with the maximum entropy change value is at the top, followed by the other records in descending order of entropy change values.

2. Generating outliers. The second step is to generate outliers, so that the entropy difference gap is determined and then compared with the max entropy gap for each value from the top of the table downwards. If the entropy difference gap is less than or equal to the max entropy gap then the algorithm continues carrying out comparisons down the table and, if the entropy difference gap is greater than or equal to the max entropy gap then the algorithm terminates and all the records up to that point are added to the outlier set $OS$. $OS$ is then displayed as the output.

This version is automated in that it requires no user input.

4 Experimentation and Results

We ran our algorithm on real-life datasets obtained from the UCI Machine Learning Repository [8] (lymphography, cancer and Australian credit card dataset).

The datasets along with their attributes are shown in table 1, table 2 and table 3. The first dataset has 148 instances with 18 attributes both numerical and categorical.

Table 1: Lymphography data set

<table>
<thead>
<tr>
<th>Case</th>
<th>Class code</th>
<th>Percentage of Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common classes</td>
<td>2,3</td>
<td>95.9%</td>
</tr>
<tr>
<td>Rare Classes</td>
<td>1,4</td>
<td>4.1%</td>
</tr>
</tbody>
</table>

Thus using the above distribution, the number of outliers in the Lymphography data set are 7.

The second dataset has 699 instances with 9 attributes all categorical.

Table 2: Wisconsin Data Set

<table>
<thead>
<tr>
<th>Case</th>
<th>Class code</th>
<th>Percentage of Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common classes</td>
<td>1</td>
<td>92%</td>
</tr>
<tr>
<td>Rare Classes</td>
<td>2</td>
<td>8%</td>
</tr>
</tbody>
</table>

Thus using the above distribution, the number of outliers in the cancer data set are 56.
The final dataset used is Australian credit card dataset which consist of 14 attributes and 2000 records. The aim of this dataset is to detect any cases of credit card theft. The data set contains a total of 2 classes. Class 1 has the largest number of instances.

<table>
<thead>
<tr>
<th>Case</th>
<th>Class code</th>
<th>Percentage of Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>No card theft</td>
<td>1</td>
<td>96%</td>
</tr>
<tr>
<td>Card Theft</td>
<td>2</td>
<td>4%</td>
</tr>
</tbody>
</table>

Thus using the above distribution, the number of outliers in the cancer data set are 80.

Now that we know the number of outliers for each dataset, AVFA is now applied on the three datasets. The results for each of the datasets are show in table 4.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>No of Outliers</th>
<th>No of Outliers detected by AVFA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lymphography</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Cancer</td>
<td>56</td>
<td>55</td>
</tr>
<tr>
<td>Australian Credit Card</td>
<td>80</td>
<td>80</td>
</tr>
</tbody>
</table>

As we can see form the above table, that AVFA for 2 datasets was able to detect 100% outliers and for the third dataset it was able to detect 98% outliers. These results clearly indicate the effectiveness of the AVFA algorithm.

5 Conclusion

Outlier detection has been a very important concept in data mining. The aim of outlier detection is to find those objects that are of not the norm. There are many applications of outlier detection from network security to detecting credit fraud. However most of the outlier detection algorithms are focused towards numerical data and do not perform well when applied to categorical data. In this paper, we proposed an automated outlier detection algorithm which specifically caters for categorical data. The results have shown that when applied on three different datasets, for two datasets it was able to detect 100% outliers and for the third dataset it detected 98% of the outliers.

References: