Resource-bounded Outlier Detection using Clustering Methods

1

Luis TORGO [a,2], and Carlos SOARES [a]
[a] *LIAAD/INESC Porto LA - FEP, University of Porto, Portugal*

Abstract.
This paper describes a methodology for the application of hierarchical clustering methods to the task of outlier detection. The methodology is tested on the problem of cleaning Official Statistics data. The goal is to detect erroneous foreign trade transactions in data collected by the Portuguese Institute of Statistics (INE). These transactions are a minority, but still they have an important impact on the statistics produced by the institute. The detectiong of these rare errors is a manual, time-consuming task. This type of tasks is usually constrained by a limited amount of available resources. Our proposal addresses this issue by producing a ranking of outlyingness that allows a better management of the available resources by allocating them to the cases which are most different from the other and, thus, have a higher probability of being errors. Our method is based on the output of standard agglomerative hierarchical clustering algorithms, resulting in no significant additional computational costs. Our results show that it enables large savings by selecting a small subset of suspicious transactions for manual inspection, which, nevertheless, includes most of the erroneous transactions. In this study we compare our proposal to a state of the art outlier ranking method (LOF) and show that our method achieves better results on this particular application. The results of our experiments are also competitive with previous results on the same data. Finally, the outcome of our experiments raises important questions concerning the method currently followed at INE concerning items with small number of transactions.

Keywords. Outlier detection, outlier ranking, hierarchical clustering, data cleaning

Introduction

This paper addresses the problem of detecting errors in foreign trade data (INTRASTAT) collected by the Portuguese Institute of Statistics (INE). The objective is to identify the transactions that are most likely to contain errors. The selected transactions will then be manually analyzed by specialized staff and corrected if an error really exists. The effort required for manual analysis ranges from simply checking the form that was submitted to a more contacts with the company that made the transaction to confirm whether the values declared are the correct ones. In any case, the process requires the involvement of expensive human resources and has significant costs to INE.

[2]Corresponding Author: Luis Torgo, LIAAD, Rua de Ceuta, 118, 6., 4050-190 Porto, Portugal; E-mail: ltorgo@inescporto.pt.

Selected transactions are usually the ones with relatively high/low values because these affect the official statistics that are published by INE the most. Therefore, this can be cast as an outlier detection problem. The goal is to detect as many of the errors as possible. However, this task is constrained by the existence of a limited amount of expensive human resources for the manual detection of errors. Additionally, the amount of human resources available for the task varies. In busier periods, these resources have to dedicate less time to this analysis while in quieter times they can do it in a more thorough way. These constraints pose interesting challenges to outlier-detection methods. Many of the methods for these detection tasks provide yes/no answers. We claim that this type of answers leads to sub-optimal decisions when it comes to manually inspecting the signalled cases. In effect, if the resources are limited we may well get more signals that we can inspect. In this case, an arbitrary decision must be done to decide which cases are to be inspected. By providing a rank of outlyingness instead, the resources can be used on the cases that have a higher probability of error. This problem occurs in many other applications, namely in fraud detection tasks.

Previous work on this problem has compared outlier detection methods, a decision tree induction algorithm and a clustering method [1]. The results obtained with the latter did not achieve the minimum goals that were established by the domain experts, and, thus, the approach was dropped. Loureiro *et al.* [2] have investigated more thoroughly the use of clustering methods to address this problem, achieving a significant boost in terms of results. Torgo [3] has recently proposed an improvement of the method described in [2] to obtain degrees of outlyingness. In this work we apply the method proposed by Torgo [3] to the INE INTRASTAT data and compare it to other alternatives.

Our method uses hierarchical clustering methods to find clusters with few transactions that are expected to contain observations that are significantly different from the vast majority of the transactions. Rankings of outlyingness are obtained by exploring the information resulting from agglomerative hierarchical clustering methods.

Our experiments with the INTRASTAT data show that our proposal is competitive with previous approaches and also with alternative outlier ranking methods.

Section 1 describes the problem being tackled in more detail as well as the results obtained previously on this application. We then describe our proposal in Section 2. Section 3 presents the experimental evaluation of our method and discusses the results we have obtained. In Section 4 we relate our work with others and finally we present the main conclusions of this paper in Section 5.

1. Background

In this section we describe the general background, including the problem (Section 1.1) and previous results (Section 1.2), that provide the motivation for this work.

1.1. Foreign Trade Transactions

Transactions made by Portuguese companies with organizations from other EU countries are declared to the Portuguese Institute of Statistics (INE) using the INTRASTAT form. Using this form companies provide information about each transaction, namely:

- Item id,

- Weight of the traded goods,
- Total cost,
- Type (import/export),
- Source, indicating whether the form was submitted using the digital or paper version of the form,
- Form id,
- Company id,
- Stock number,
- Month,
- Destination or source country, depending on whether the type is export or import, respectively.

At INE, the data are inserted into a database. Figure 1 presents an excerpt of a report produced with data concerning import transactions from 1998 of item with id 101, as indicated by the field labeled "NC", below the row with the column names.[3]

Trade with EU countries - Detailed Declaration
IMPORT (1998)

0 R	F L U	N LOTE	N FORM	N OPERATOR	M O N	N TRA	CNT	WEIGHT (KG)	COST (kPTE)	COST/WEIGHT (PTE/KG)
NC = 101										
2	1	1008	010240	00000000	01	01	005	005	1 820	4 064	2 233
2	1	1060	011778	00000000	02	01	001	005	694 830	2 189	3
2	1	1076	012252	00000000	03	01	003	005	873	1 546	1 770
2	1	1127	013791	00000000	04	01	011	005	4 760	10 415	2 188
2	1	1086	012553	00000000	05	01	006	005	3 908	724	185
TOTAL FOR ITEM									706 191	18 938	

Figure 1. An excerpt of the INTRASTAT database. The data were modified to preserve confidentiality.

Errors often occur in the process of filling forms. For instance, an incorrectly introduced item id will associate a transaction with the wrong item. Another common mistake is caused by the use of incorrect units like, for instance, declaring the weight in tons instead of kilos. Some of these errors have no effect on the final statistics while others can affect them significantly.

The number of transactions declared monthly is in the order of tens of thousands. When all of the transactions relative to a month have been entered into the database, they are manually verified with the aim of detecting and correcting as many errors as possible. In this search, the experts try to detect unusual values on a few attributes. One of these attributes is Cost/Weight, which represents the cost per kilo and is calculated using the values in the Weight and Cost columns. In Figure 1 we can see that the values for Cost/Weight in the second and last transactions are much lower than in the others. The corresponding forms were analyzed and it was concluded that the second transaction is, in fact, wrong, due to the weight being given in grams rather than kilos, while the last one is correct.

The goal of this project is to reduce the time spent on this task by automatically selecting a subset of the transactions that includes almost all the errors that the experts

CS: inserted back because it it informative and enables us to have another

[3]Note that, in 1998, the Portuguese currency was the *escudo*, PTE.

would detect by looking at all the transactions. According to INE experts, to be minimally acceptable the system should select less than 50% of the transactions containing at least 90% of the errors. However, as stated earlier, given that human resources are quite expensive, the smaller the number of transactions, the better. Additionally, the same people are involved in other tasks in INE and sometimes are not available to evaluate INTRASTAT transactions. Therefore, the number of transactions that can be manually analyzed varies over different months.

Finally, we note that computational efficiency is not important because the automatic system will hardly take longer than half the time the human expert does.

1.2. Previous Results

Different approaches were tried on this problem. Several months worth of transaction from 1998 and 1999 were used. The data were provided in the form of two files per month, one with the transactions before being analyzed and corrected by the experts, and the other obtained after that process. The integration of the information from the two files proved much harder than could be expected. Some of the problems found were:

- difficulty in determining the primary key of the tables, even with the help of the experts;
- some transactions existed in one of the files but not in the other;
- incomplete information, sometimes because it was not filled in the forms, others due to the reporting software (e.g., values below a given threshold were considered too low and not printed in the report).

Some of the problems were handled by eliminating the corresponding records, while others were simply ignored because they were not expected to affect the data significantly. This meant that, as it is common in data mining projects, most of the time was spent in data preparation [4].

Four very different methods were applied. Two come from statistics and are univariate techniques: box plot [5] and Fisher's clustering algorithm [6]. The third one, Knorr & Ng's cell-based algorithm [7], is an outlier detection algorithm which, despite being a multivariate method, was used only on the Cost/Weight attribute. The last is C5.0 [8], a multivariate technique for the induction of decision trees.

Although C5.0 is not an outlier detection method, it obtained the best results. This was achieved with an appropriate transformation of the variables and by assigning different costs to different errors. As a result, 92% of the errors were detected by analyzing just 52% of the transactions. However, taking advantage of the fact that C5.0 can output the probability of each case being an outlier, the transactions were ordered by this probability. Based on this ranking of transactions in terms of their probability of being an error, it was possible to detect 90% of the errors by analyzing the top 40% of the transactions.

The clustering approach based on Fisher's algorithm was selected because it finds the optimal partition for a given number of clusters of one variable. It was applied to all the transactions of an item, described by a single variable, Cost/Weight. The transactions assigned to a *small cluster*, that is, a cluster containing significantly fewer points than the others, were considered outliers. The distance function used was Euclidean and the number of clusters was $k = 6$. A small cluster was defined as a cluster with fewer points than half the average number of points in the k clusters. The method was applied to data

relative to two months and selected 49% of the transactions which included 75% of the errors, which did not accomplish the goals set by the domain experts.

Further work based on clustering methods was carried out by Loureiro *et al.* [2], who have proposed a new outlier detection method based on the outcome of agglomerative hierarchical clustering methods. Again, this approach used the size of the resulting clusters as indicators of the presence of outliers. The basic assumption was that outlier observations, being observations with unusual values, would be distant (in terms of the metric used for clustering) from the "normal" and more frequent observations, and therefore would be isolated in smaller clusters. In [2], several settings concerning the clustering process were explored and experimentally evaluated on the INTRASTAT problem. The best setup met the requirements of human experts (inspecting less than 50% of transactions enabled finding more than 90% of the errors), by detecting 94.1% of the errors by inspecting 32.7% of the transactions. In spite of this excellent result, the main drawback of this approach is the fact that it does not allow a control over the amount of inspection effort we have available. For instance, if 32.7% is still too much for the human resources currently available we face the un-guided task of deciding which of these transactions will be inspected. The work presented on this paper tries to overcome this practical limitation.

2. Hierarchical Clustering for Outlier Ranking

As discussed above, outlier-detection problems with constraints on the amount of resources that limit the maximum number of selected cases can better be handled by providing a ranking of the examples in terms of their expected level of outlierness. The use of rankings allows the users to select the number of transactions to inspect according to the available human resources, with a guarantee that the results for that working point are "optimal", at least according to the outlier-ranking method.

Clustering algorithms can be used to identify outliers as a side effect of the clustering process (e.g. [9]). Most clustering methods rely on a distance metric and thus can be seen as distance-based approaches to outlier detection [7]. However, iterative methods like hierarchical clustering algorithms (e.g. [10]) can also handle different density regions, which is one of the main drawbacks of distance-based approaches. In effect, if we take agglomerative hierarchical clustering methods, for instance, they proceed in an iterative fashion by merging two of the current groups (which initially are formed by single observations) based on some criterion that is related to their proximity. This decision is taken locally, that is for each pair of groups, and takes into account the density of these two groups only. This merging process results in a tree-based structure usually known as a dendrogram. The merging step is guided by the information contained in the distance matrix of all available data. Several methods can be used to select the two groups to be merged at each stage. Contrary to other clustering approaches, hierarchical methods do not require a cluster initialization process that would inevitably spread the outliers across many different clusters thus probably leading to a rather unstable approach. Based on these observations we have explored hierarchical clustering methods for detecting both local and global outliers [2].

In this paper we present an approach that takes advantage of the dendrogram generated by hierarchical clustering methods to produce a ranking of outlyingness. This ap-

proach was first described in [3] and is also based on agglomerative clustering methods. Informally, the idea behind our proposal is to use the height (in the dendrogram) at which any observation is merged into a group of observations as an indicator of its outlyingness. If an observation is really an outlier this should only occur at later stages of the merging process, that is the observation should be merged at a higher level than "normal" observations. More formally, we set the outlyingness factor of any observation as,

$$OF_H(x) = \frac{h}{N} \tag{1}$$

where h is the level of the hierarchy H at which the case is merged,[4] and N is the number of training cases (which is also the maximum level of the hierarchy by definition of the hierarchical clustering process).

One of the main advantages of our proposal is that we can use a standard hierarchical clustering algorithm to obtain the OF_H values without any additional computational cost. This means our proposal has a time complexity of $O(N^2)$ and a space complexity of $O(N)$ [11]. We use the *hclust()* function of the statistical software environment R [12], which is based on Fortran code by F. Murtagh [13]. This function includes in its output a matrix (called **merge**) that can be used to easily obtain the necessary values for calculating directly the value of OF_H according to Equation 1.

Figure 2.(a) shows an artificial data set with two marked clusters of observations with very different density. As it can be observed there are two clear outliers: observations 1 and 12. While the former can be seen as a global outlier, the latter is clearly a local outlier. In effect, it is only regarded as an outlier because of the high density of its neighbors, as it is in effect nearer observation 2 than, say the 14th from the 15th. However, as these two latter are in a less compact region their distance is not regarded as a signal of outlyingness. This is a clear example of a data set with both global and local outliers and we would like our method to clearly signal both 1 and 12 as observations with a high probability of being outliers.

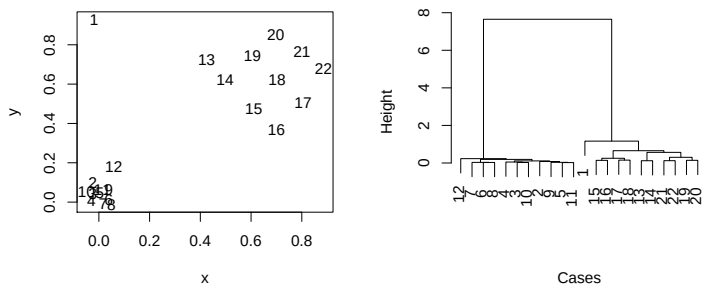

Figure 2. An artificial example.

[4]Counting from bottom up.

Table 1. Outlier ranking for the example of Figure 2.

Rank	CaseID	OF_H
1	1	0.9091
2	12	0.6818
3	17	0.5909
4	18	0.5909
5	19	0.5455

Figure 2.(b) shows the dendrogram obtained by using an agglomerative hierarchical clustering algorithm. As it can be seen, both 1 and 12 are the last observations to be individually merged into some cluster. As such, it does not come as a surprise that when running our method on this data we get the top 5 outliers shown on Table 1.

In spite of this success, this method has serious problems when facing compact groups of outliers. In effect, if we have a data set where there are a few outliers that are very similar to each other, they will be merged with each other very quickly (i.e., at a low level of the hierarchy) and thus will have a very low OF_H value despite being outliers. Figure 3 illustrates the problem. For this data set, the method ranks observations 9 and 10, which are clear outliers, as the least probable outliers (they are in effect the first to be merged). This problem is particulary important in our application and also in fraud detection. In both cases, it is often true that the interesting observations are not completely isolated from all the others. They sometimes stem from a behavior which, although rare, is systematic (e.g., a company always declares transactions in counts rather than in kilos).

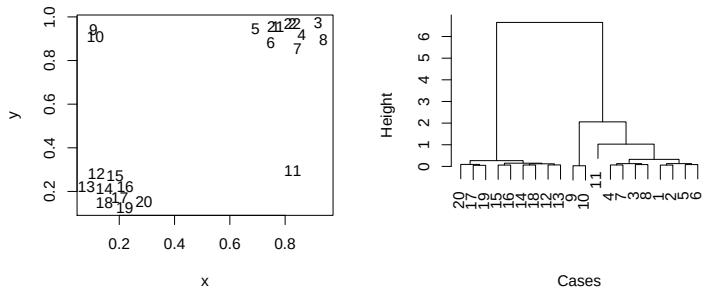

Figure 3. An artificial example that is problematic for our initial proposal.

The example of Figure 3 shows a clear failure of our initial proposal. The failure results from considering only the height at which individual observations are merged and not groups of observations. When there is a small group of similar observations that is quite different from others, such that it could make sense to talk about a set of outliers, they will only be merged with other groups at later stages but they will merge with each other very early in the process. Therefore, our proposal will not consider this as a sign of outlyingness of the members of that group. Still, the general idea of our proposal remains valid so we need to generalize it for these situations. We can do this by assigning a value similar to that of Equation 1 to all members of the smallest group of any merge that

Table 2. Outlier ranking for the example of Figure 3 using our new proposal.

Rank	CaseID	OF_H
1	9	0.8100
2	10	0.8100
3	11	0.8075
4	15	0.6300
5	16	0.6300

occurs along the hierarchical clustering process. However, we should reinforce this value with some size-dependent factor (i.e., the smaller the group, the more probable that its elements are outliers). Formally, for each merge of a group g_s with a group g_l, where $|g_s| < |g_l|$, we set the outlier factor of the members of g_s as,

$$OF\left(g_s\right) = \begin{cases} 0 & \text{if } |g_s| > t \\ \left(1 - \frac{|g_s|}{N}\right) \times \frac{h}{N} & \text{if } |g_s| < t \end{cases} \tag{2}$$

where $|g_s|$ is the cardinality of the smallest group, g_s, t is a threshold that indicates the number of observations above which a group can not be regarded as a set of outliers for the data set, and h is the level of the hierarchy where the merge occurs. The OF value of the larger group g_l is set to zero. The value of OF ranges from zero to one, and it is maximum when a single observation is merged at the last level of the hierarchy.

Any observation can belong to several groups along its upwards path through the dendrogram. As such, it will probably get several of these scores at different levels. We set the outlyingness factor of any observation as the maximum OF score it got along its path through the dendrogram. By proceeding this way we are in effect enabling the method to detect local outliers, which at some merging stage might have got a very high score of OF because they are clear outliers with respect to some group that they have merged with, even though at higher levels of the hierarchy (i.e., seen more globally), they might not get such high OF values. This means that the outlyingness factor of an observation is given by

$$OF_H(x) = \max_{g \in G_x} OF(g) \tag{3}$$

where G_x is the set of groups in the dendrogram to which x belongs.

Applying this method to the problematic example of Figure 3, we get the outlier ranking shown in Table 2. This is the expected result for this problem, which indicates that this new formulation is able to handle compact and small groups of outlier observations, like for instance observations 9 and 10 of this problem.

3. Experimental Evaluation

This section describes a series of experiments designed with the goal of checking the performance of our method on the INTRASTAT data set. We have compared our OF_H

method with our previous approach [2] and also with the state of the art in terms of obtaining degrees of outlyingness: the LOF method [14].

The INTRASTAT data set has some particularities that lead to an experimental methodology that incorporates some of the experts' domain knowledge so that the methodology better meets their requirements.

We start by describing the measures used to assess the quality of the results (Section 3.1), then we discuss the experimental setup (Section 3.2), the algorithms that were tested (Section 3.3) and finally we discuss the results (Section 3.4).

3.1. Evaluation Measures

In order to evaluate the validity of the resulting methodology we have taken advantage of the fact that the data set given to us had some information concerning erroneous transactions. In effect, all transactions that were inspected by the experts and were found to contain errors, were labeled as such. Taking advantage of this information we were able to evaluate the performance of our methodology in tagging these erroneous transactions for manual inspection. The experts were particularly interested in two measures of performance: *Recall* and *Percentage of Selected Transactions*, which are discussed next.

Recall ($\%R$) can be informally defined in the context of this domain as the proportion of erroneous transactions (as labeled by the experts) that are selected by our models for manual inspection. Ideally, our models should select a set of transactions for manual inspection that included all the transactions that were previously labeled by the experts as errors. However, taking into consideration the difficulty of the problem, INE experts established the value of 90% as the minimum acceptable recall.

Regarding the percentage of selected transactions ($\%S$) this is the proportion of all the transactions that are selected for manual inspection by the models. This statistic quantifies the savings in human resources achieved by using the methodology: the lower this value the more manual effort is saved. INE experts defined 50% as the maximum admissible value for this statistic. Given the fact that our method outputs a ranking of outlyingness we can easily control the value of this measure. The user can decide which percentage of transactions he/she wants to check and then use the ranking provided by our method to select the transactions corresponding to the selected percentage. Given that 50% is the maximum value and that it is important to release human resources for other tasks and that the available resources vary, in our experimental evaluation, we have collected results for four different percentage of selected transactions: 35%, 40%, 45% and 50%. All of these settings satisfy the requirements established by the experts concerning this measure.

An important issue that must be taken into account when analyzing the value of recall ($\%R$) is the quality of the labels assigned to transactions. When a transaction is labeled as an error, this classification is reliable because it means that the experts have analyzed the transaction and found an error. However, since not all transactions are analyzed, there may be some that are labeled as "normal" but are, in fact, errors. Many of these are transactions that were actually detected by the experts but, because they are not expected to affect the trade statistics which are computed based on these data, are not corrected. However, it is possible that some significant errors are missed by the experts. Here, we will not address this issue and simply focus on selecting the errors that were detected by the domain experts.

Table 3. The "base" results just for including the items with less than 10 transactions.

	Jan/1998	Feb/1998	Mar/1998	May/1998	Jun/1998	Aug/1998	Sep/1998	Oct/1998
%S	35.7	30.8	27.7	24.5	32	21.0	17.0	22.5
%R	35.4	40.4	38.7	29.7	37	30.8	25.4	27.9

3.2. Experimental Setup

According to INE experts, the items should be inspected separately due to the rather diverse distribution of the prices of the products. For instance, the variation of values for rice is smaller than for heavy machinery. As such we have applied our algorithm to the set of transactions of each item in turn.

Our outlier ranking method is designed for multivariate analysis. However, following another suggestion from the domain experts we have focused our study of the INTRASTAT data set in a single variable, $Cost/Weight$. Domain experts give particular attention to this variable as they believe it is the most efficient variable for detecting the important errors.

Given that INE processes the data on a monthly basis we have decided to use this very same logic in our tests. This methodology will also enable us to compare our results with the results obtained in [1], where the same strategy was followed.

One final constraint has an important effect on the results. According to INE experts, all items with very few transactions, referred to as *infrequent items*, must be set for manual inspection. This reduces the number of transactions that the outlier detection methods may, in fact, select. The domain experts defined 10 as the minimum number of transactions required for an item to be classified as infrequent. As shown in Table 3, this fact alone has a big impact on the process. The number of transactions that can be selected by the outlier detection method is not 50%, as originally established, but ranges from 15% to 35% (approx.). Furthermore, the concentration of errors in the infrequent items are generally higher than in the others but not that much higher. In the selected transactions, the number of errors found represents between 25% and 40% (approx.) of all the errors. Considering, for instance, the month of Jan/1998, the items with less than 10 transactions represent 35.7% of all the transactions and contain 35.4% of the errors. This means, that, to achieve the target of 90% of Recall, the outlier detection method needs to find almost 55% of the errors by selecting less than 15% of the transactions, to stay within the maximum effort tolerated by INE experts, which is 50%.

The experimental methodology that we have used is better described by Algorithm 1. This algorithm calculates the value of the Recall for each month of the testing period, given a certain desired human effort (given by a provided $\%S$).

3.3. Algorithms

Using Algorithm 1 we have collected the performance, in terms of *Recall*, of our proposed method and also of the LOF method.

The clustering-based outlier detection method proposed here (Section 2) has several parameters. The first is the agglomeration method used with the *hclust()* function. In our experiments we have tested several alternative: the ward, single, complete, average, mcquitty, median and centroid methods. Another parameter of our method is the distance

Algorithm 1. The experimental methodology.

Require: $D, PercS$ ▷ D is the data set, $PercS$ is the %S selected by the user
Ensure: $\%R$ ▷ The vector of $\%R$'s for each month

for all $m \in Months$ **do**
  $TotTrans \leftarrow \|D_m\|$
  $TotErrors \leftarrow \|\{tr \in D_m : \text{label}(tr) = error\}\|$
  $TotInsp \leftarrow TotRight \leftarrow 0$
  $OFs \leftarrow \phi$ ▷ Will contain the outlying factors of all candidate trans.
  for all $i \in ITEMS$ **do**
    if $\|\{tr \in D_{i,m}\}\| < 10$ **then**
      $TotInsp \leftarrow TotInsp + \|\{tr \in D_{i,m}\}\|$
      $TotRight \leftarrow TotRight + \|\{tr \in D_{i,m} : \text{label}(tr) = error\}\|$
    else
      $OFs \leftarrow OFs \cup OutlierRanking(D_{i,m})$
    end if
  end for
  $R \leftarrow PercS \times TotTrans - TotInsp$ ▷ The $\%S$ remaining...
  if $R > 0$ **then**
    $TotInsp \leftarrow TotInsp + R$
    $OFs \leftarrow SortDecreasing(OFs)$
    $TotRight \leftarrow TotRight + \|\{tr \in \{OFs\}_{n=1}^{R} : \text{label}(tr) = error\}\|$
  end if
  $\%R_m \leftarrow TotRight/TotErrors$
end for

function used. For this parameter we have experimented with both the euclidean and camberra functions. Finally, our method also requires the specification of a limit on the size of a group in order to be selected as a group of (potential) outliers (the t threshold in Equation 2). The possible combinations of these settings makes up for a total of 14 variants of our method.

With respect to LOF, we have used the implementation of this algorithm that is available in the R package **dprep** [15]. We have also experimented with 14 variants of this method, namely by varying the number of neighbours used by the method from 2 to 28 in steps of 2.

In our graphs of results we also plot the $\%S$ and $\%R$ value of the method described in [2], which is denoted in the graphs as "LTS04". This method is not an outlier ranking algorithm. It simply outputs the (unordered) set of transactions it judges as being outliers, which leads to a single pair of $\%S$ and $\%R$ values. In this case the user is not able to adjust the $\%S$ value to the available resources. By chance, in none of the testing months the 50% limit of selected transactions was surpassed but with this type of methods there is not such guarantee. In months when the available resources are not sufficient to analyze all the transactions selected, the experts must decide which ones to let aside. Additionally, in the months when the number of transactions that could be analyzed by the available resources is greater than the number of selected transactions, the experts must arbitrarily

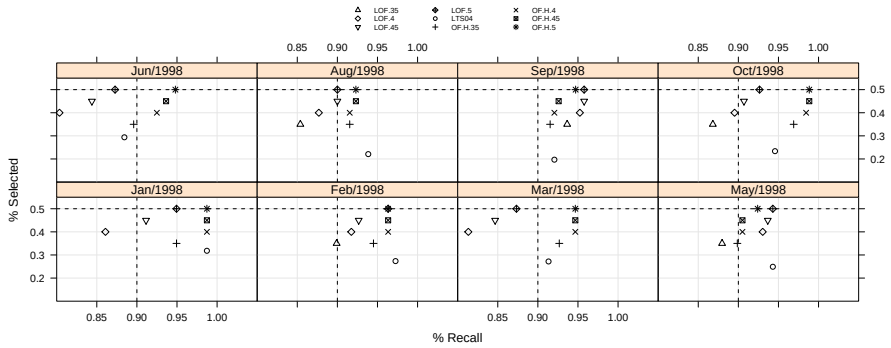

Figure 4. The results of the experiments on the INTRASTAT data.

select further transactions to check. For the "LTS04" method the same 14 variants used with OF_H were tried.

3.4. Results

Figure 4 shows the results of our comparative experiments in terms of recall ($\%R$) and percentage of selected transactions ($\%S$) for each of the 8 available testing months. For each of the methods we have always reported the best result of the 14 variants that were tried. These can thus be regarded as the best possible outcome of these methods. Each graph in the figure represents a month. All graphs have two dotted lines indicating the experts requirements (at least 90% recall and at most 50% selected transactions). This means that for each graph the best place to be is the bottom right corner (maximum $\%R$ and minimum $\%S$). Still, the most important statistic is Recall as long as we do not overcome the 50% limit. The four points for both OF_H and LOF represent the four previously selected working points in terms of $\%S$. Still, we should recall that both methods would be better represented by lines as any other working points could have been selected. Some of the points are not shown on some graphs because the respective method achieved a very poor score that is outside of the used axes limits.

The results of our experiments (cf. Figure 4) clearly indicate that our method is competitive with a state of the art outlier ranking method, LOF. This confirms previous results on a different set of applications [3]. Moreover, our method is always able to fulfil the minimum requirement of 90% recall, which is not always the case with LOF. Compared to "LTS04", both OF_H and LOF lose a few times in terms of achieving the same $\%R$ for the same level of $\%S$. Still, we should recall that "LTS04" provides no flexibility in terms of available human resources and thus it can happen (as for instance in Jun/1998) that the solution provided by this method does not attain the objectives of the experts or even that it is not feasible because it requires too many resources.

CS: which ones? is this the right figure?

As discussed in Section 3.2, the results presented in Figure 4 include all transactions from infrequent items, i.e, items with less than 10 transactions. An analysis of Figure 4 taking into account the impact of infrequent items (cf. Table 3), raises an important question. In effect, the decision of inspecting infrequent items was "imposed" by the INE experts. However, by looking at our results we think this decision is rather questionable. For instance, in Jan/1998 the inclusion of the small items incurred in a "cost" of

CS: why not plot a line connecting the variants of each method?

$\%S = 35.7\%$, whilst only allowing us to detect 35.4% of the errors. By simply adding 10% more transactions, our method ($OFh.45$) was able to boost the recall to 95%. Now the question is: is it really necessary to analyze all the transactions in infrequent items? The small amount of data makes the outlier detection method proposed here inappropriate for these items. However, it may is possible to use some other form of statistical decision method to reduce the amount of transactions from infrequent items to analyze. Our results clearly indicate that statistical-based outlier detection methods are able to do a much better job than this brute force approach. Therefore, if we can reduce the amount of effort required for infrequent items, then more resources can be dedicated to analyzing transactions selected by the outlier detection method proposed here.

A lesson that can be learned from this observation is that not all domain-specific knowledge is useful. However, addressing the problem of using automatic methods to select transactions in infrequent items is not just a technical challenge, caused by the small volume of data. If we are able to successfuly detect outliers in these items, the next challenge will be to convince the experts to change their beliefs.

4. Related Work

Outlier detection is a well studied topic (e.g. [16]). Different approaches have been taken to address this task. Distribution-based approaches (e.g. [17,18]) assume a certain parametric distribution of the data and signal outliers as observations that deviate from this distribution. The main drawbacks of these approaches lie on the constraints of the assumed distributions. Depth-based methods (e.g. [19]) are based on computational geometry and compute different layers of k-d convex hulls and then represent each data point in this space together with an assigned depth. In practice these methods are too inefficient for dealing with large data sets. Knorr and Ng [7] introduced distance-based outlier detection methods. These approaches generalize several notions of distribution-based methods but still suffer from several problems, namely when the density of the data points varies (e.g. [14]). Density-based local outliers [20,14] are able to find this type of outliers and are the appropriate setup whenever we have a data set with a complex distribution structure. These authors defined the notion of Local Outlier Factor (LOF) for each observation, which naturally leads to the notion of outlier ranking. The key idea of this work is that the notion of outlier should be "local" in the sense that the outlier degree of any observation should be determined by the clustering structure in a bounded neighborhood of the observation. In Section 3 we have seen that our method compares favorably with the LOF algorithm on the problem of detecting errors in portuguese foreign trade transactions.

Other authors have looked at the problem of outliers from a supervised learning perspective (*e.g.* [1,21]). Usually, the goal of these approaches is to classify a given observation as being an outlier or as a "normal" case. These approaches are typically affected by the problem of unbalanced classes that occurs in outlier detection applications, because outliers are, by definition, much less frequent than the "normal" observations. If adequate adjustments are not made, this kind of class distribution usually deteriorates the performance of the supervised models [22].

5. Conclusions

In this paper we have presented a method for obtaining a ranking of outlyingness using an hierarchical clustering approach. This method uses the height at which cases are merged in the clustering process as the key factor for obtaining a degree of outlyingness.

We have applied our methodology to the task of detecting erroneous foreign trade transactions in data collected by the Portuguese Institute of Statistics (INE). The results of the application of our method to this problem clearly met the performance criteria outlined by the human experts. Moreover, our results outperform previous approaches to this same problem. Compared to these previous approaches, our method provides a result that allows a flexible management of the available human resources for the manual task of inspecting the potential erroneous transactions.

Our results have also revealed a potential inefficiency on the process used by INE to handle the items with a small number of transactions. In future work we plan to address these items in a way that we expect to further improve our current results.

Acknowledgements This work was partially funded by FCT (Programa de Financiamento Plurianual de Unidades de I&D and project Rank! - PTDC/EIA/81178/2006).

CS: check if special ack section exists

References

[1] C. Soares, P. Brazdil, J. Costa, V. Cortez, and A. Carvalho. Error detection in foreign trade data using statistical and machine learning methods. In N. Mackin, editor, *Proc. of the 3rd International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, pages 183–188, 1999.

[2] A. Loureiro, L. Torgo, and C. Soares. Outlier detection using clustering methods: a data cleaning application. In Malerba D. and May M., editors, *Proceedings of KDNet Symposium on Knowledge-based Systems for the Public Sector*, 2004.

[3] L. Torgo. Resource-bounded fraud detection. In Neves et. al, editor, *Proceedings of the 13th Portuguese Conference on Artificial Intelligence (EPIA'07)*, LNAI, pages 449–460. Springer, 2007.

[4] Shichao Zhang, Chengqi Zhang, and Qiang Yang and. Data preparation for data mining. *Applied Artificial Intelligence*, 17(5 & 6):375 – 381, May 2003.

[5] J.S. Milton, P.M. McTeer, and J.J. Corbet. *Introduction to Statistics*. McGraw-Hill, 1997.

[6] W.D. Fisher. On grouping for maximum homogeneity. *Journal of the American Statistical Association*, 53:789–798, 1958.

[7] Edwin M. Knorr and Raymond T. Ng. Algorithms for mining distance-based outliers in large datasets. In *Proceedings of 24rd International Conference on Very Large Data Bases (VLDB 1998)*, pages 392–403. Morgan Kaufmann, 1998.

[8] R. Quinlan. *C5.0: An Informal Tutorial*. RuleQuest, 1998. http://www.rulequest.com/see5-unix.html.

[9] R. Ng and J. Han. Efficient and efective clustering method for spatial data mining. In *Proc. of VLDB'94*, 1994.

[10] L. Kaufman and P.J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York, 1990.

[11] F. Murtagh. Complexities of hierarchic clustering algorithms: state of the art. *Computational Statistics Quarterly*, 1:101–113, 1984.

[12] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, 2008. ISBN 3-900051-07-0.

[13] F. Murtagh. Multidimensional clustering algorithms. *COMPSTAT Lectures 4, Wuerzburg: Physica-Verlag*, 1985.

[14] M. M. Breunig, H. P. Kriegel, R. Ng, and J. Sander. Lof: Identifying density-based local outliers. In *Proceedings of ACM SIGMOD 2000 International Conference on Management of Data*, 2000.

[15] Edgar Acuna and members of the CASTLE group. *dprep: Data preprocessing and visualization functions for classification*, 2008. R package version 2.0.

[16] Victoria Hodge ; Jim Austin. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22:85–126, 2004.

[17] D. M. Hawkins. *Identification of Outliers*. Chapman and Hall, 11 New Fetter Lane, London EC4P 4EE, 1980.

[18] V. Barnett and T. Lewis. *Outliers in statistical data*. John Wiley, 1994.

[19] F. Preparata and M. Shamos. *Computational Geometry: an introduction*. Springer-Verlag, 1988.

[20] M. M. Breunig, H. P. Kriegel, R. Ng, and J. Sander. Optics-of: Identifying local outliers. *Lecture Notes in Computer Science*, 1704:262–270, 1999.

[21] L. Torgo and R. Ribeiro. Predicting outliers. In N. Lavrac, D. Gamberger, L. Todorovski, and H. Blockeel, editors, *Proceedings of Principles of Data Mining and Knowledge Discovery (PKDD'03)*, number LNAI in 2838, pages 447–458. Springer, 2003.

[22] G. Weiss and F. Provost. The effect of class distribution on classifier learning: an empirical study. Technical Report ML-TR-44, Department of computer science, Rutgers University, 2001.