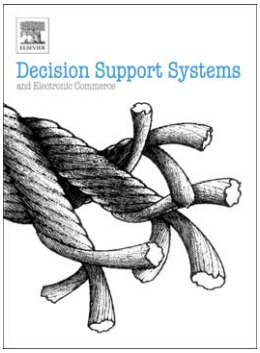


# Algorithm for the detection of outliers based on the theory of rough sets

Francisco Maciá-Pérez, José Vicente Becrna-Martínez, Alberto Fernández-Oliva, Miguel Abreu-Ortega

PII: S0167-9236(15)00086-X
DOI: doi: 10.1016/j.dss.2015.05.002
Reference: DECSUP 12609

To appear in: *Decision Support Systems*

Received date: 12 March 2014
Revised date: 12 March 2015
Accepted date: 3 May 2015

Please cite this article as: Francisco Maciá-Pérez, José Vicente Becrna-Martínez, Alberto Fernández-Oliva, Miguel Abreu-Ortega, Algorithm for the detection of outliers based on the theory of rough sets, *Decision Support Systems* (2015), doi: 10.1016/j.dss.2015.05.002

**Algorithm for the Detection of Outliers Based on the Theory of Rough Sets**

Francisco Maciá-Pérez[1,a] , José Vicente Berna-Martínez[1,b], Alberto Fernández-Oliva[2,c], Miguel Abreu-Ortega[2,d]

[1] Department of Computer Technology. University of Alicante. Carretera San Vicente del Raspeig s/n - 03690 San Vicente del Raspeig – Alicante

[2] Department of Computer Science, School of Mathematics and Computer Science. University of Havana

[a]pmacia@dtic.ua.es, [b]jvberna@dtic.ua.es, [c]afdez@matcom.uh.cu, [d]miguel87@lab.matcom.uh.cu

**Corresponding author:**

Jose Vicente Berna-Martinez

jvberna@dtic.ua.es

Tel.: +34 965 90 3400 ext. 1307

Fax: +34 96 590 9643

**Keywords**

Knowledge Discovery, Detection of Outliers, Rough Set Theory

**Abstract**

Outliers are objects that show abnormal behavior with respect to their context or that have unexpected values in some of their parameters. In decision-making processes, information quality is of the utmost importance. In specific applications, an outlying data element may represent an important deviation in a production process or a damaged sensor. Therefore, the ability to detect these elements could make the difference between making a correct or an incorrect decision. This task is complicated by the large sizes of typical databases. Due to their importance in search processes in large volumes of data, researchers pay special attention to the development of efficient outlier detection techniques. This article presents a computationally efficient algorithm for the detection of outliers in large volumes of information. This proposal is based on an extension of the mathematical framework upon which the basic theory of detection of outliers, founded on Rough Set Theory, has been constructed. From this starting point, current problems are analyzed; a detection method is proposed, along with a computational algorithm that allows the performance of outlier detection tasks with an almost–linear complexity. To illustrate its viability, the results of the application of the outlier–detection algorithm to the concrete example of a large database are presented.

## 1.- Introduction

Decision support systems are computer-based programs that assist decision makers in effective and efficient decision-making. The proper functioning of these systems requires large amounts of precise, high-quality data. However, if the data contain abnormal, unrealistic or simply erroneous elements, it may misguide the decision-making process, thereby leading to incorrect

results. These abnormal elements must be detected, isolated and analyzed to check whether they have any real meaning or simply represent a monitoring error. Currently, data sets in the real world and their environments present a wide range of difficulties that limit the efficiency of some existing detection methods. One of the most noteworthy problems is data sets can be very large and dynamic, imposing the need for efficient algorithms with regard to time complexity. Recent investigations related to Knowledge Discovery in Databases (KDD) have paid special attention to issues regarding the detection of outliers, which become more serious with the large volumes of information stored in today's databases [1, 2]. If, in general, KDD-Data mining (KDD-DM) processes are directed toward the discovery of representative behavioral patterns, the detection of outliers takes advantage of the high marginality of these objects, (marginality refers to how much or little different which is an element from the rest), and they are detected by measuring their degree of deviation with respect to the aforementioned patterns. From the perspective of KDD-DM, outlier detection can be viewed in two different ways: outliers can be considered undesirable objects that must be treated or eliminated at the stage of preparation of the data, as their presence in the set may interfere with the efficient detection of trustworthy patterns [3]; or they can be considered as objects that should be identified for their implicit relevance to the processing of the data [2]. In the latter case, they must not be eliminated from the data set, because for some applications, outliers are more representative and interesting than common events from the point of view of information discovery. Therefore, outlier detection is also a process of information (knowledge) discovery and is of great utility for the analysis and interpretation of data.

The range of applicability of outlier detection is very wide and diverse, and applications can be found in widely varied scenarios. In wireless networks, outlier allows for the detection of atypical readings and their subsequent correction [4]. By monitoring activities of various kinds, such as the activity of a mobile phone or an online store, it is possible to detect suspicious or unlawful activities [5]. In the study of DNA irregularities, outlier detection may lead to the discovery of genetic alterations that result in disease and structural defects [6]. In the automated control of assembly lines, it allows the detection of production defects [7]. In pharmaceutical research, it aids in the identification of new molecular structures [8]. This diversity in the range of application is one of the justifying motives for the wide variety of existing methods of outlier detection. The data in each situation possess a distinct nature and definition space, and therefore, the detection methods must adjust to the data types and contexts where they will be applied [9]. Therefore, the search for efficient methods that may be utilized in any situation, which are more flexible, adaptable and scalable, is a problem of great interest.

Some of the techniques being applied efficiently in KDD-DM processes are related to Rough Set Theory [10, 11], which is advantageous due to its flexibility and adaptability to different scenarios. This adaptability is demonstrated by the variety of related works found in the literature, including the process of evaluation of Complex Information Systems [12], learning in neural networks [13], analysis of Reconfigurable Manufacturing Systems (RMS) [14], solutions to problems in the field of investment [15], application to the Grid Scheduling Process [16], adjudication of bank credits [17], image processing [18], and the evaluation of business innovation capabilities [19]. Thus, the ability of these techniques to model a wide range of real–world situations, their efficiency in the resolution of various types of problems, and their wide range of applicability have been made manifest.

The use of rough sets (RS) extends within the KDD-MD and is also beginning to be utilized as the foundation for the characterization and detection of outliers. This approach is a novel point of view with great potential that shows promise for the construction of efficient algorithms [20,21], capable of detecting outliers with a high degree of marginality. However, these detection schemes have as a disadvantage the inconvenience of using the concept of non–

redundant exceptional sets to classify the most contradictory elements of a data set, from which the outliers are obtained. The weak point of this scheme is that identifying such sets requires the identification of a power set, which leads to a problem of exponential time complexity ($\Omega(2^n)$, $n$ being the cardinality of the data set). In today's world, where data volumes are of great size, this problem makes such schemes unfeasible from a computational point of view, even though they may be formally sound in mathematical rigorous terms.

The present article analyzes the problem of exponential temporal complexity exhibited by current algorithms based on rough set theory [20]. We propose an expansion of the existing mathematical framework, which allows for the creation of a method of outlier detection based on rough sets that is in fact computationally viable, with a corresponding algorithm of almost linear time complexity. Furthermore, a runtime study of the use of the proposed algorithm applied to a realistic data set is presented, showing that, indeed, it shows in practice the behavior that is described mathematically.

The remaining sections of the article have been organized as follows: Section 2 summarizes the current state of the technique, along with some background for this research work, relevant aspects of RS theory, and its main inconveniences. In Section 3, we discuss the foundations of rough set theory and present a simple example to clarify its inner workings. In Section 4, the expansion of the mathematical framework is developed, which will subsequently allow the proposal of the computational algorithm. In Section 5, an algorithm is proposed for the detection of outliers based on the basic model of rough sets, emphasizing specific aspects of its implementation. In Section 6, the different tests that were performed with the proposed algorithm on a real data set are shown, corroborating the theoretical results. Finally, in Section 7, the main conclusions of this work are presented, along with the main lines of future research. In the Appendix at the end of this work, the theoretical framework proposed in Section 4 is explicitly demonstrated.

## 2.- Background

Generally speaking, in data mining, the detection of outliers allows for the identification of unexpected input in a database and, on these grounds, the determination of various types of errors, data usage fraud, the existence of valid but atypical values, and many other features of interest. Therefore, its applications are highly diverse, for example, the detection of intruders in computer networks [22]; the data mining of manuscripts in the context of a project for the digitalization of the cultural and scientific heritage of Bulgaria [23]; applications in medical diagnosis, where outlier detection can aid in the diagnosis of a given pathology [24]; the detection of outliers in climate studies related to the temperature ranges in different world cities [25]; the analysis and processing of population data for the US Census Bureau's Income report [26, 27, 28]; the detection of traffic risks based on which accident prevention measures can be taken [29]; outlier detection techniques in data sets of credit card usage information, employed to detect their misuse [30]; outlier detection used for the adequate classification of crystals based on chemical and physical tests [30]; in the context of sports, outlier detection used to monitor the performance of NBA players in the USA [31]; and in video surveillance, outlier detection allows guaranteed safety in public areas (video/image data mining) [32]. As can be appreciated, outlier detection is a subject of applicability as vast as the existing types of databases.

Due to the great number of scenarios and data types, different approaches to the problem of outlier detection have arisen, and above all, proposals that address large data volumes have begun to gain importance. This problem was treated first in the field of statistics: statistical

models are generally appropriate for the processing of data sets with quantitative, real, continuous values, or at least qualitative data with ordinal values. Nowadays there is an ever-increasing need for the processing of categorical (non–ordinal) data. This requirement considerably limits the applicability of statistical methods. Another deficiency of statistical methods is their limited functionality in high–dimensionality (multivariable) spaces, where it is generally exceedingly difficult to find adequate models [33].

There are methods based on non–parametric approximations, among which distance–based outlier detection methods can be found. One of the most widely utilized is the method of k–nearest neighbors (K-NN) [34, 35]. There are different approaches to the K-NN algorithm, but all use a metric that is appropriate for the calculation of distances between neighbors, such as the Euclidean distance or the Mahalanobis distance. There are also proposals that optimize the basic K-NN algorithm [36].

In general, it can be said that there are numerous techniques for the detection of outliers in which algorithms of different kinds are combined [20, 37]. Among the most outstanding methods, some categories can be identified, such as methods based on distributions [38], depth [39], distances [34, 36], densities [40], clusters [41], or support vectors [42]. Most recent research works address various detection methods based on artificial intelligence techniques, fundamentally, techniques related to machine learning [43]. In [44], we find a very complete compilation of the most outstanding outlier detection methods. Most of the distance–based methods are of at least quadratic of time completion order with respect to the number of elements in the data set, which may be unacceptable if the data set is very large or dynamic. On a different front, statistical methods essentially center on the detection of outliers among single–variable data. They require *a priori* knowledge of the data distribution. In these cases, the user must model the data utilizing a statistical distribution, and the outliers are determined depending on how they appear in relation to the postulated model. The main problem with this approach lies in the number of possible situations and on the possibility that the user may lack sufficient knowledge of the data distribution.

Considering that no universally applicable outlier detection approach is available and that researchers must focus their efforts on the selection of an acceptable method for their specific data set, this subject still poses a very open problem. A consequence is the continued appearance of new models and new methods based on a diversity of schemes and approaches to the problem at hand. One of these new propositions is the application of Rough Set Theory to outlier detection, where previous studies [20] and the results and achievements already attained in this line of research [45] serve as the main precedents.

Rough Set Theory [11] is an extension of Set Theory for application to the case of incomplete or insufficient information. This theory arises from the practical need to solve classification problems, and it assumes that with every object in the universe there is associated a certain amount of information: the existing knowledge about the object, expressed in terms of the values of some set of properties that describe it. This theory has the added appeal of a simple and solid mathematical foundation: the theory of equivalence relations, which here allows for the description of partitions constituted by indiscernible classes that group objects of similar attributes, that is, a data classification methodology.

The successful application of Rough Set Theory in multiple contexts demonstrates its efficiency and versatility for the solution of a variety of problems. In particular, it has been applied with outstanding results in KDD-DM processes. Examples include the following: in the field of trading systems, the theory has been used for prediction purposes [46]; in the field of machine learning, through the conception of classification algorithms based on decision trees [47]; in

basic research on the development of intelligent systems [48]; in classification problems, through the use of decision trees [49]; and in the field of bioinformatics [50].

### 3.- Rough Set Theory Groundwork

Although in [11], one can find a full explanation of the mathematical foundations of Rough Set Theory, the following definitions are reiterated to lay the groundwork for our work.

Let $U \neq \phi$ be the (finite) universe, and let $r \subseteq UXU$ be some equivalence relation defined over *U*.

Let $U / r$ be the set of equivalence classes induced by $r$ in *U*.

Let $X \subseteq U$ be a set of elements that satisfy a given *concept* (in other words, the elements of a study set that meet with characteristics). Two approximations are defined that characterize *X*:

Upper approximation: $\bar{r}(X) = \cup\{Y \in U / r : Y \cap X \neq \phi\}$. The union of all equivalence classes induced by *r* in *U* whose intersection with *X* is not empty. These are the equivalence classes induced by $r$ in *U* that contain some elements that satisfy the *concept*.

Lower approximation: $\underline{r}(X) = \cup\{Y \in U / r : Y \subseteq X\}$. The union of all equivalence classes induced by *r* in *U* that are contained in *X*. These are the equivalence classes induced by *r* in *U* where all of the elements satisfy the *concept*.

Rough Set theory itself defines the concept of the *boundary* as $BN(X) = \bar{r}(X) - \underline{r}(X)$. These are the equivalence classes induced by *r* in *U* that each contain some element that satisfies the *concept* and also some element that does not.

Using these definitions, in [20] the mathematical characterization of outliers is developed. This characterization is focused on the concept of the Inner Boundary. This work proposes a series of mathematical concepts whose main elements are collected below. These definitions lead to the formalization of the concept of a non–redundant exceptional set. This set defines the exceptional elements, which therefore are classified as outliers.

**Definition 1 – Inner boundary:** Let $\Re = \{r_1, r_2, ..., r_m\}$ be *m* equivalence relations defined over the universe *U*. The inner boundary of *X* with respect to $r_i$ is defined as follows:
$B_i(X) = BN_i(X) \cap X = X - r_i(X)$

In other words, the inner boundary is the intersection of the concept *X* with the elements of the boundary. These elements represent a type of contradiction in *X* because in its equivalence class, there are some elements that satisfy *X* and others that do not. Figure 1 shows the abstract ideas of the universe *U* partitioned according to the equivalence relation *r*, the concept *X* contained within *U*, the upper and lower approximations, the boundary, and the inner boundary.

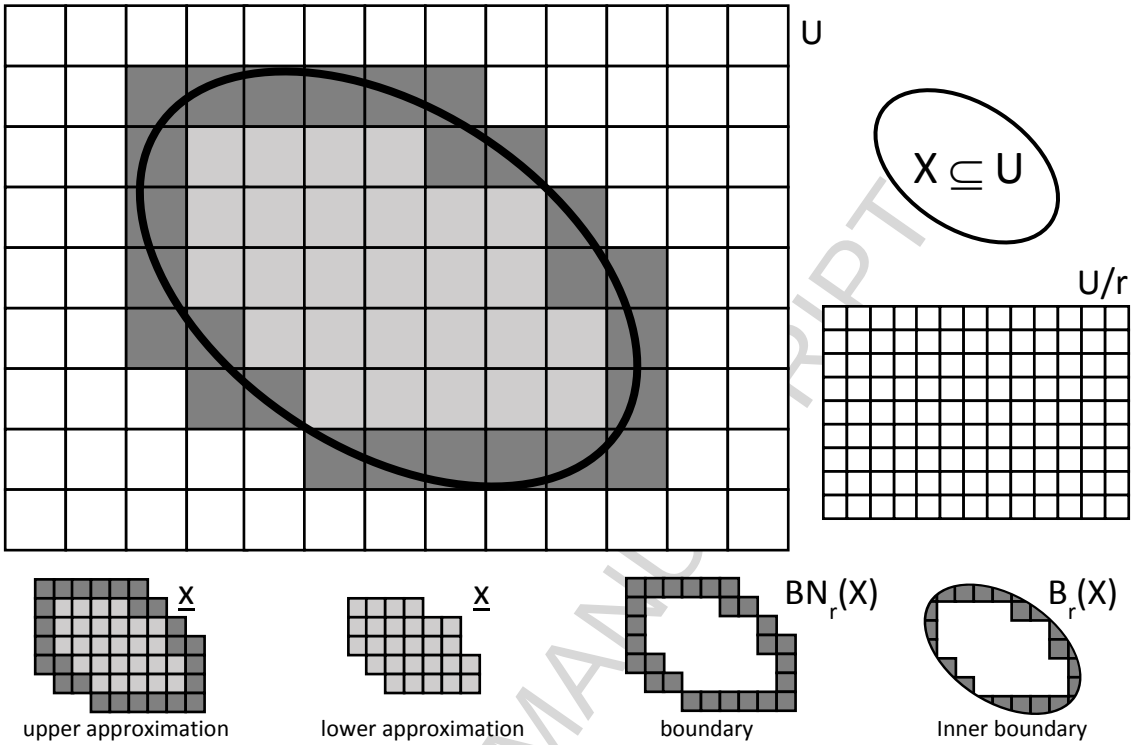


**Figure 1**. Illustration of the universe U partitioned according to r, the concept X in U, the upper and lower approximations, the boundary and the inner boundary.

**Definition 2 – Exceptional set:** Let $e \subseteq X$ such that: $\forall r_i \in \Re$, $B_i(X) \neq \phi$, and $e \cap B_i(X) \neq \phi$. The set *e* is called an exceptional set of X with respect to $\Re$.

Exceptional sets are made out of elements that contradict all of the equivalence relations that characterize *U*.

**Definition 3** – **Dispensable element:** Let *e* be an exceptional set of $X$ with respect to $\Re$, and let $x \in e$ such that $e-\{x\}$ is also exceptional with respect to $\Re$. It is then said that *x* is dispensable in *e* with respect to $\Re$. In the opposite case, *x* is **indispensable** in *e* with respect to $\Re$.

Dispensable elements can be eliminated from an exceptional set such that the exceptional set still represents all of the contradictions that characterize *U*.

**Definition 4 – Non–redundant exceptional set:** An exceptional set is said to be non–redundant if all its elements are indispensable. These sets contain elements that are contradictory according to all of the equivalence relations that characterize *U*, and all of these sets are representative of at least one contradiction. In addition, these sets constitute the fundamental source of elements to be considered *outlier* candidates in subsequent stages of the detection method.

**Definition 5 – Degree of marginality**: Let *x* be an arbitrary element of *X*. The *degree of marginality* of *x* with respect to $\Re$ is the number of different inner boundaries of *X* with respect to $\Re$ that contain *x*: $BD(X) = \left|\{B_i(X), i = 1,2,...,m : x \in B_i(X)\}\right|$.

The degree of marginality represents the *degree of contradiction* of an element with respect to the equivalence relations that characterize *U*.

**Definition 6 – Degree of exceptionality:** The degree of exceptionality of *x* is defined as follows: $OD(x)=BD(x) / |\Re|$

This concept serves to *normalize* the degree of marginality of an element such that it can be limited to values between 0 and 1.

**Definition 7 - Outlier:** An outlier in *X* with respect to $\Re$ is an object *x* that belongs to some non–redundant exceptional set of *X* with respect to $\Re$ and that has a degree of exceptionality larger than some given threshold $\mu$.

### 3.1.- Example of the search for outliers in a data set

This subsection shows a simple example that exposes the inner workings of the proposed algorithm. This example strives to show the functional aspects of the proposal, not to validate it. In Section 5, we examine a realistic case based on a large data set, which does serve as validation of the proposal.

In this example, the universe *U* represents 21 patients. **Table 1** shows the data for this universe. In the database, a diagnosis is established for each patient based on their temperature and on whether or not they have a headache, to determine whether they have a cold.

**Table 1**. Example data that represent the universe *U*.

| ID | Headache | Temperature | Diagnosis |
|---|---|---|---|
| 1 | Yes | Normal | Unknown |
| 2 | No | Very high | Cold |
| 3 | Yes | High | Cold |
| 4 | No | Normal | Unknown |
| 5 | Yes | Very high | Cold |
| 6 | No | High | Unknown |
| 7 | No | High | Sunstroke |
| 8 | No | Very high | Cold |
| 9 | Yes | Normal | - |
| 10 | Yes | Normal | Sunstroke |
| 11 | Yes | Very high | Cold |
| 12 | No | Normal | - |
| 13 | Yes | Normal | Headache |
| 14 | Yes | Normal | Headache |
| 15 | No | High | Sunstroke |
| 16 | No | Very high | Cold |
| 17 | No | Very high | Cold |
| 18 | No | Normal | - |
| 19 | No | Very high | Cold |
| 20 | Yes | High | Cold |
| 21 | Yes | High | Unknown |

Two equivalence relations that will be considered in the analysis are defined in Figure 2. Each of these relations partitions the universe *U* into a given number of equivalence classes.

$$r_1 = \left\{ x \in U: \left\{ \begin{array}{l} 1 \text{ yes\_headache}(x) \\ 0 \text{ otherwise} \end{array} \right\} \right\}$$

$$r_2 = \left\{ x \in U: \left\{ \begin{array}{l} 0 \text{ yes\_normal\_temperature}(x) \\ 1 \text{ yes\_high\_temperature}(x) \\ 2 \text{ otherwise} \end{array} \right\} \right\}$$

**CONCEPT** $C = \{x \in U \wedge \text{cold}(x)\}$

**Figure 2.** Equivalence relations that will be used in the analysis and definition of the concept to be searched.

Starting from the equivalence relation $r_1$, in Figure 3, the equivalence classes formed in the universe *U* are shown. In this case, in both classes there exist elements that satisfy the concept and elements that do not; therefore, both classes lie inside the inner boundary of *C* with respect to $r_1$. The elements of both classes that satisfy the concept form the inner boundary.

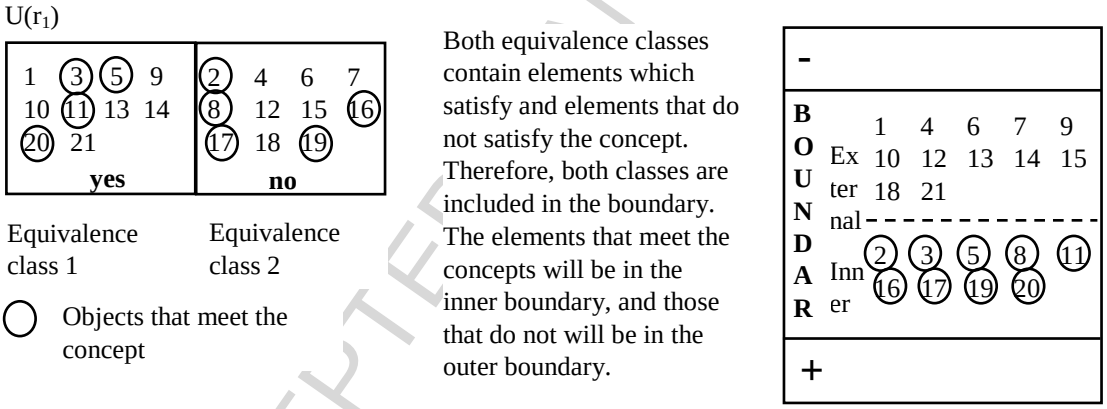


**Figure 3**. Equivalence classes in the universe *U* starting from $r_1$ and the definition of the boundary in terms of whether the classes satisfy the concept.

On the other hand, Figure 4 shows the equivalence classes that belong to the partition of *U* from $r_2$. In this case, equivalence class 1 does not contain elements that satisfy the concept, and thus, the class is irrelevant for the analysis. Class 2 has elements that satisfy the concept and elements that do not. The elements that satisfy the concept belong to the inner boundary. On the other hand, all elements of class 3 meet the concept, and thus this class is completely included in the lower approximation of $r_2$.

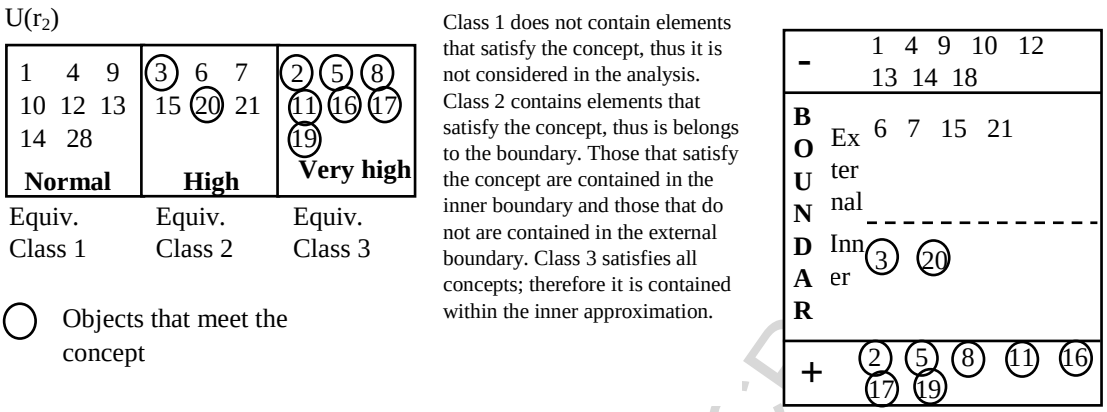


**Figure 4**. Equivalence classes in the universe *U*, from $r_2$, and the definition of the boundary in terms of whether the classes satisfy the concept or not.

The following inner boundaries are obtained:

$B_1 = \{2, 3, 5, 8, 11, 16, 17, 19, 20\}$

$B_2 = \{3, 20\}$

The calculated set E that contains all elements of U that belong to some non–redundant exceptional set is as follows:

$E = \{3, 20\}$

The degree of exceptionality (number of inner boundaries to which it belongs, divided by the total number of inner boundaries) for each element of E is as follows:

Degree_of_Exceptionality(3) = 2/2 = 1

Degree_of_Exceptionality (20) = 2/2 = 1

Bearing in mind that the value of the degree of exceptionality is a value between 0 and 1, it can be stated that both elements would be considered outliers for any given threshold μ larger than 0. This fact can be interpreted as both elements being in contradiction with element 21, which has the same symptoms but is nevertheless not diagnosed as a COLD.

### 3.2.- Conclusions regarding the temporal complexity of the present proposal

The definitions given in previous section demonstrate that for determining whether an element is an outlier in *X* with respect to $\Re$ , it is necessary to first evaluate all non-redundant exceptional sets along with their degree of exceptionality. A simplified approximation of the proposed detection method can be summarized in two steps:

- Step one: determine all non-redundant exceptional sets on the given set *X* (a concept).
- Step two: detect the outliers from the elements of the non-redundant exceptional sets. Every element whose degree of exceptionality is higher than a given threshold μ will be considered an outlier.

If it is assumed that all outliers in *X* must belong to non-redundant exceptional set, then, if any element in *X* does not meet this condition, we can state that such an element is not an outlier.

The proposed detection method is simple with respect to the theoretical foundation on which it is based, but its computational implementation from the given definition of an *outlier* is an intractable problem. This is because it is required to determine all subsets *X* (the power set of *X*)

and evaluate whether they are redundant. From a mathematical point of view, it is well known that the cardinality of the power set is $\Omega(2^{|X|})$. Its construction is therefore computationally unfeasible because it produces algorithms of exponential temporal complexity.

As mentioned above, the large sizes of current databases, which contain hundreds of millions of data records, do not allow for the use of current algorithms based on rough set theory. Thus, a method to reduce the order of temporal complexity of the algorithm must be developed before such a rough-set-theory-based algorithm can be used.

The results of [20] provide a theoretical framework for the detection of outliers based in rough sets, but without a concrete solution, remaining within the bounds of the theoretical. The Rough Set Model [11] has been successfully applied to solve a vast number of problems, which demonstrates its efficiency and versatility. As a consequence, the proposal of [20]— being based on the aforementioned model— is expected to be powerful and efficient when used in data mining problems where its application is feasible. One of the elements that may limit the efficiency of a method of *outlier* detection is the nature of the data on which the method is implemented. The method proposed in the coming sections is applicable to both continuous and discrete (categorical) data and does not pose any limitations on the size of the data set.

**4.- Expansion of the mathematical framework**

To conceive the design and implementation of a computationally efficient algorithm for outlier detection, we propose an expansion of the theoretical framework presented above. The following lemmas and propositions constitute the mathematical foundation of the outlier detection algorithm to be proposed. This expansion complements the definition of an exceptional set such that the use of a power set is not necessary for the construction of an algorithm for outlier detection, and one can instead resort to other means to characterize an element as an outlier.

Let *C* be the *concept*, and let $\Re = \{r_1, r_2, ..., r_m\}$ be a set of *equivalence relations* defined over a finite data *universe U*. Let $X \subseteq U$ be the set of elements of *U* that meet *C*, and let $B_i$ be the *inner boundary* of *X* with respect to $r_i$.

**Proposition 8:** $\forall e, e' \subseteq X$, if $e$ is exceptional and $e \subseteq e'$, then $e'$ is also exceptional. In other words, if *e* is an exceptional set, any set that contains *e* is also exceptional. In the example shown in Section 3.1, element 3 belongs to both inner boundaries. Therefore, the set {3, 4} is exceptional (it has elements from all inner boundaries). Thus, any set that contains all elements of {3} is also exceptional: {3, 4, 10} {3, 4, 5}.

**Lemma 9**: Let *f* be a non–redundant *exceptional set*, and let *a*: $a \in X \wedge a \in f, \exists f \Leftrightarrow \exists i, 1 \leq i \leq m$ such that $B_i^c \cup \{a\}$ is an *exceptional set* (where $B^c$ is a boundary of the concept c). That is, given that *f* belongs to a non-redundant exceptional set, there exists some inner boundary such that its complement, the union element *a*, is exceptional. In the example presented in Section 3.1, the set {3} is non-redundant exceptional (it has elements of all inner boundaries, and all of its elements are indispensable).

Therefore, $B_1^c \cup \{3\} = \{1, 4, 6, 7, 9, 10, 12, 13, 14, 15, 18, 21, 3\}$ is also an exceptional set.

**Lemma 10:** If $\exists\, a \in B_i$, $1 \leq i \leq m$, such that $B_i^c \cup \{a\}$ is not an *exceptional set*, then $\exists\, j$, $1 \leq j \leq m$, $j \neq i$, such that $B_j \subset B_i$.

In other words, if element $a$ belongs to an inner boundary and the union of the said element with the complement of the boundary is not exceptional, then there exists another inner boundary that is a subset of it.

Let us take the inner boundary $B_1$ in the example presented in Section 3.1. Then, $B_1^c \cup \{5\} = \{1, 4, 6, 7, 9, 10,12,13,14, 15, 18,21, 5\}$. This set is not exceptional (it does not contain elements of the inner boundary $B_2$). Thus, there exists some inner boundary that is its subset. In this example, it is easy to note that $B_2 \subseteq B_1$.

**Corollary 11:** If $\forall\, j,\ 1 \leq i, j \leq m, j \neq i$, and the condition that $B_j \not\subset B_i$ is met (which means that the *inner boundary* $B_i$ does not completely contain any other *inner boundary*), then $\forall\, a \in B_i,\ B_i^c \cup \{a\}$ is an *exceptional set*.

In the example presented in Section 3.1, let us take as a reference the inner boundary $B_2$. It is easy to see that $B_1 \not\subset B_2$.,Thus, the following holds:

$B_2^c \cup \{3\}$ and $B_2^c \cup \{20\}$ are exceptional sets (the elements 3 and 20 belong to all inner boundaries).

**Lemma 12:** Let $a \in$ X. If $\exists\, j, j \neq i,\ 1 \leq i, j \leq m$, such that $B_j \subseteq B_i$ and $B_i^c \cup \{a\}$ is an *exceptional set*, then $B_j^c \cup \{a\}$ is also an *exceptional set*.

In the example presented in Section 3.1, $B_2 \subseteq B_1$. Then,
$B_1^c \cup \{3\} = \{1, 4, 6, 7, 9, 10,12,13,14, 15, 18,21, 3\}$ is an exceptional set (element 3 belongs to both inner boundaries), and thus,

$B_2^c \cup \{3\} = \{1,2,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,21, 3\}$ is also an exceptional set.

**Definition 13:** Let $f$ be any non–redundant *exceptional set*. The set $E_i,\ 1 \leq i \leq m$, is defined as follows: $E_i = \{a: a \in X, a \in f, f \cap B_i = \{a\}\}$. $E_i$ will contain all the elements of $X$ that belong to some non–redundant *exceptional set* (all taken into account) and that are also members of the *inner boundary* $B_i$.

Starting from this point, one can reach the following conclusion: $E = \bigcup_{i=1}^{m} E_i$ , which is the set of all elements of $X$ that belong to some non–redundant *exceptional set*.

**Lemma 14:** $\forall i,\ 1 \leq i \leq m$, the condition is met that $E_i \subseteq B_i$, that is, all the elements of some particular $E_i$ are elements of the *inner boundary* $B_i$. This result is trivial and stems directly from the definition of the set $E_i$.

**Lemma 15:** Let $a \in X,\ 1 \leq i \leq m$. $B_i^c \cup \{a\}$ is an *exceptional set* if and only if $a \in E_i$.

In the example presented in Section 3.1, if we take the inner boundary $B_1$, we have the following:

$B_1^c \cup \{3\} = \{1, 4, 6, 7, 9, 10,12,13,14, 15, 18,21, 3\}$. This set is exceptional; however, $B_1^c \cup \{5\} = \{1, 4, 6, 7, 9, 10,12,13,14, 15, 18,21, 5\}$ is not because element 3 belongs to $E_1$, whereas element 5 does not.

All of these results have been formally demonstrated. For clarity throughout the text, the formal demonstrations are placed in the Appendix. From these mathematical foundations, in the following section, the proposed algorithm and the mathematical aspects of the theoretical framework on which it is based are presented in detail.

**5.- Algorithm for the detection of outliers based on the basic Rough Set Model**

Before presenting the actual detection algorithm, let us establish a few preliminaries.

Let *X* be the *concept* to be considered, and let $\Re = \{r_1, r_2, ..., r_m\}$ be a set of equivalence relations (*criteria*) defined over *U*.

The inner boundaries ($B_i$) of *X* are calculated with respect to every element of $\Re$. For every element that satisfies *X* and that also belongs to any inner boundary $B_i$, its *degree of marginality* is calculated.

**Input**: *U*: Universe, *X*: *Concept*, *R*: The set of all equivalence relations.
**Output**: *B*: inner boundaries, regions of the rough set model.
**Initializations**: $\overline{X} = \{\}$ $\underline{X} = \{\}$ $\forall r \in \Re : B_r = \{\}$

```
1  for each r ∈ R                          // For each equivalence relation  <r>
2    Pr = CLASSIFY-ELEMENTS (U, r)         // Pr is the partition induced by the equivalence
                                           relation r
3    for each class ∈ Pr                   // For each equivalence relation induced by r
4       if class ∩ X = class               // All elements of the class meet the concept
5         X = X ∪ class                    // The set X is updated.
6       else if class ∩ X ≠ ϕ              // There exist elements that meet the concept,
                                           and others that do not.
7         X̄ = X̄ ∪ class                    // The set X̄ is updated.
8         B_r = B_r ∪ (class ∩ X)           // The set B_r is updated.
```

**Algorithm 1.** Algorithm for the construction of the main regions of the rough sets model.

Once this process is concluded, the set *E* is constructed from the elements of the sets $E_i$. If the analysis of a particular set $E_i$ detects some element of *C* that had previously been identified as a member of some other set $E_j$ and therefore had already been included in *E*, this element will not be considered.

As a theoretical assumption, the algorithm considers only different, not empty, inner boundaries. There is no loss of generality, because if there are two identical boundaries, the elements in them that belong to some non–redundant exceptional set would be the same in both cases, and therefore having a duplicate does not contribute anything of relevance to the results. The empty inner boundaries are not taken into account by definition (**Definition 2**—Exceptional set).

The essence of the algorithm is to determine the inclusion relation between inner boundaries and, based on these relations, to make decisions and perform actions that are simply direct applications of one of the Lemmas and Corollaries presented and demonstrated above, which serve as the theoretical framework for the algorithm itself. Finally, the set *E* is obtained, containing all elements of the universe *U* that belong to some non–redundant exceptional set. The following Algorithm 1 is a pseudo–code version of the proposed detection algorithm.

**Input**: U: Universe, X: Concept, R: Set of equivalence relations, β: Classification error, μ: Exceptionality threshold, B: Inner boundaries.
**Output**: E: outliers
**Initialization**: OUTLIERS = {}, E = {}

```
1   BUILD-REGIONS (U, X, R, β)
2   for each r ∈ R                          // For every equivalence relation <r>
3     existsBoundary = FALSE                // No inner boundary exists that is a subset of B_r^β
4       for each q ∈ R                      // For each equivalence relation <q> different from <r>
5         if r == q
            continue
6         if B_q ⊂ B_r                      // If the inner boundary of q is a subset of the inner
                                            // boundary  of <r>, then its elements are ruled out as
                                            // members of the set  of possible OUTLIERS: E
7           existsBoundary = TRUE
8         break                             // No need to continue
9       if NOT existsBoundary               // If no inner boundary is a subset of the one being
                                            // analyzed,
                                            // then all elements of the inner boundary of r make up
                                            // the set of possible OUTLIERS: E
10        E = E ∪ B_r^β
11  for each e ∈ E                          // Impose a condition on each element of E given the
                                            // exceptionality threshold
12    if EX-DEGREE(e) ≥ μ
13      OUTLIERS = OUTLIERS ∪
    {e}
```

**Algorithm 2.** Algorithm for the construction of the set OUTLIERS

As demonstrated and thanks to the expansion of the mathematical framework, it is no longer necessary to determine all subsets of the universe *U*. The algorithm works on the elements included in the inner boundaries. As demonstrated below, the implications of this simplification are transcendental for the method of calculation because it changes the order of the temporal complexity of the problem from $2^n$ (exponential) to *n x m*, which can be considered linear. *n* is the cardinality of the universe U, and *m* is the number of equivalence relations.

This change in the order of the temporal complexity allows for the application of rough set theory to very large data sets. In addition, as stated above, some outlier detection methods are limited by the nature of the data that they can handle (*e. g.*, continuous data, discrete data, or a mixture of both). The mathematical framework utilized here uses equivalence relations that do not suffer from this type of limitation because the elements of the data set are perfectly classified within an equivalence class, given the equivalence relation $r_i$.

This algorithm is utilized for the computational implementation of a tool that allows the application of the method described. Below, the details of its implementation are described.

### 5.1.- Considerations regarding the computational implementation

The following are the input parameters of the algorithm: the universe *U*, the concept *C*, the relations $r_i$, $1 \leq i \leq m$ - $\Re = \{r_1, r_2, ..., r_m\}$, and the established degree of exceptionality (*threshold* value $\mu$).

The basic data structure utilized in the algorithm is the *dictionary* structure, where by dictionary is understood a set of pairs (*key*, *value*) where the *key* is an arbitrary object to which one and only one object of the type *value* is associated.

In the algorithm, the *keys* are obtained as a result of applying a classifier to a given element of the universe. This classifier is associated with a particular equivalence relation $r_i$, with $1 \leq i \leq m$, and it allows the classification of the members of the equivalence classes, as defined by the equivalence relation. The *values* associated with the *keys* are lists of elements that belong to the equivalence class identified by the *key* associated with the given *value*.

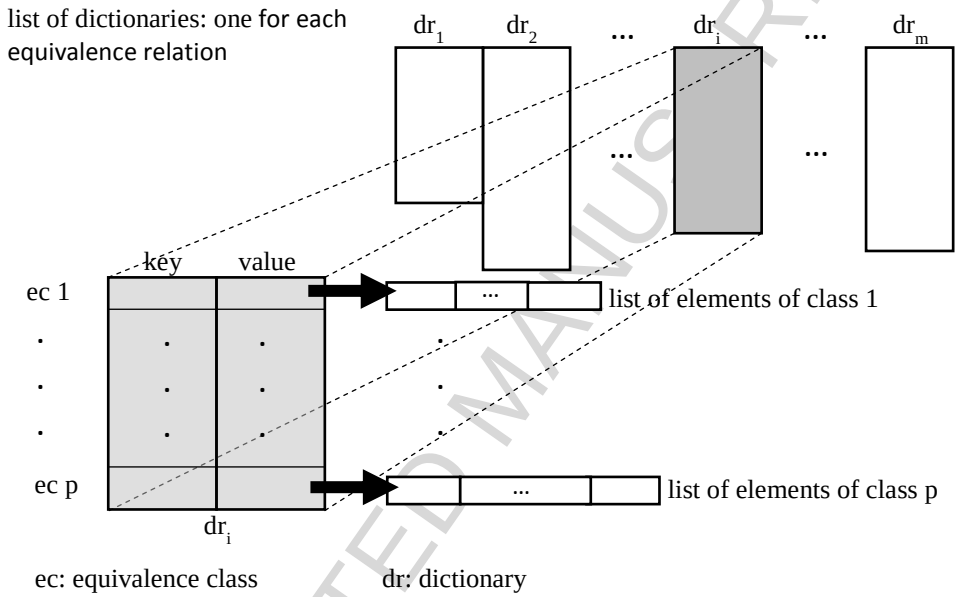


**Figure 5**. Data structures utilized

Figure 1 shows that, for each equivalence relation, a dictionary is built and a list of dimension *m* is formed, where m is the number of equivalence relations considered. The input to the algorithm is stored as lists. A list is a basic type, and therefore, it is not necessary to give any details regarding its functionality.

Following the example of Section 3.1, the data structure is determined by the following dictionaries:

$dr_1$

| | |
|---|---|
| ec1 | 1 3 5 9 10 11 13 14 20 21 |
| ec2 | 2 4 6 7 8 12 15 16 17 18 19 |

$dr_2$

| | |
|---|---|
| ec1 | 2 4 9 10 12 13 14 |
| ec2 | 3 6 7 15 20 21 |
| ec3 | 2 5 8 11 16 17 19 |

**Figure 6**. Data structures utilized in the example presented in Section 3.1.

To each equivalence relation corresponds a dictionary, and each dictionary is formed by equivalence classes that produce the application of the relation on the *U*.

According to the data structures utilized, it can be said that the spatial complexity of the algorithm is *O(n x m)*, where *n* is the cardinality of the universe, and *m* is the number of equivalence relations considered in the analysis, as each dictionary (each region formed from an equivalence relation) may contain at most all the elements of the universe. The amount of memory occupied by the rest of the data structures does not exceed this order of magnitude.

The computational implementation of the algorithm consists of two fundamental stages:

**Stage 1 – Formation of inner boundaries**: the classifiers are applied (one per equivalence relation) to the elements of the universe (data set) with the goal of forming the inner boundaries for each of the equivalence relations considered in the analysis. This operation is carried out in the algorithm by the routine BUILD-REGIONS (U, X, R, β). The time complexity of this operation is $O(n \times m \times c)$, where $c$ is the cost of classifying each element.

**Stage 2 – Process of outlier detection**: concrete implementation of the rest of the proposed algorithm. Time complexity $O(n \times m^2)$.

Considering stages 1 and 2, the execution time for the full algorithm is estimated as $O(max(O(stage\ 1), O(stage\ 2))) = O(stage\ 2) = O(n \times m^2)$.

In general, the number of equivalence relations involved in the analysis in the vast majority of cases is not very large in relation to the number of rows in the table. Therefore, the quadratic dependence of the execution time on the number of equivalence relations does not greatly affect the algorithm's execution time, as we will be able to verify using a realistic case. As will be shown in the results, this quadratic dependence is almost linear for small values of $m$ ($m \leq 20$).

The calculation of the degree of exceptionality of an element in the universe is relatively simple, once the inner boundaries have been calculated. It lies in determining— for each element in the universe— the number of inner boundaries to which it belongs and finding the ratio of this number to the total number of inner boundaries. This task is performed by the EX-DEGREE(e) function.

### 5.2.- Classification error

One particularly critical topic in classification systems is the occurrence of classification errors, which are known as false positive/negatives. The classification of a non-exceptional element as exceptional is called a false positive, whereas the failure to classify an exceptional element as exceptional is called a false negative.

In our case, classification errors can occur in two places of the algorithm: during the formation of the inner boundaries (algorithm 1) and during the construction of the set of outliers (algorithm 2).

Regarding the first point, because we use equivalence relations that partition the universe into equivalence classes, the inclusion of elements that do not belong to the equivalence class is out of the question, assuming that the relations have been appropriately defined. The opposite case—non-inclusion of an element that does belong to the equivalence class—is likewise out of the question.

Regarding the point about the construction of the set of outliers, however, the algorithm uses as the degree of exceptionality EX-DEGREE(e) to determine when an element belongs to this set. The degree of exceptionality, which results from dividing the number of inner boundaries to which the element belongs by the total number of existing equivalence relations, is a number between 0 and 1. In the algorithm, the threshold μ can be tuned; that is, we can vary the value that determines whether an element is exceptional enough to be considered an outlier, e.g. the elements with a value $\mu > 0{,}6$ are outliers. As a consequence, elements with degrees of exceptionality below this arbitrary threshold will not be classified as outliers. We will demonstrate how variation of the threshold modifies the detection of outliers.

## 6.- Tests and validation

In this section, a realistic case of the proposal is demonstrated over a widely used database. The tests that were performed had the fundamental goal of validating the time complexity of the algorithm on the basis of the theoretical analysis of this parameter and of measuring the detection quality.

It was decided that a data set from the UCI Machine Learning Repository, from the Center for Machine Learning and Intelligent Systems of the University of California, Irvine [44], would be used for the tests. The UCI Machine Learning Repository offers a collection of databases (data sets) that are used by the scientific community for research on topics related to Machine Learning and Data Mining for the empirical analysis of algorithms. In particular, the tests were performed using a data set from this site that contains data extracted from the Census Bureau Database of the USA [26], including 48,842 instances with 14 attributes that mix continuous and categorical data. Explicit references to more than 50 articles where this data set is utilized are shown on the website of the UCI Machine Learning Repository. On this website [44], the most outstanding characteristics of the data set are noted, and a detailed explanation of its attributes can be found.

The hardware on which the results were validated has the following characteristics: 1.5 GHz INTEL Pentium 4 CPU with 256 MB of RAM. Platform: Windows XP SP3. The goal of these tests was the empirical observation of the theoretical conclusions; among them, that the execution of the algorithm produces results with linear complexity. For this reason, it was decided not to utilize specialized or high–performance hardware.

### 6.1.- Execution time

The tests were performed considering the variation of all parameters that define the size of the input to the algorithm. That is, the size of the data set, the number of columns and the number of equivalence relations were considered in the analysis.

Figure 7 shows the results of executing the algorithm on the data set. For this test, 30,000 rows were utilized, and the number of columns considered in the analysis was varied from 5 to 14. Figure 7 also shows the corresponding execution times. The results presented in this figure lead to the conclusion that the dimensionality of the data set (number of columns in the table) does not influence the execution time. It can be seen that an increase in dimensionality does not represent a problem for the correct execution of the method. The results of the execution times corroborate the theoretical analysis for the calculation of the time complexity of the algorithm.

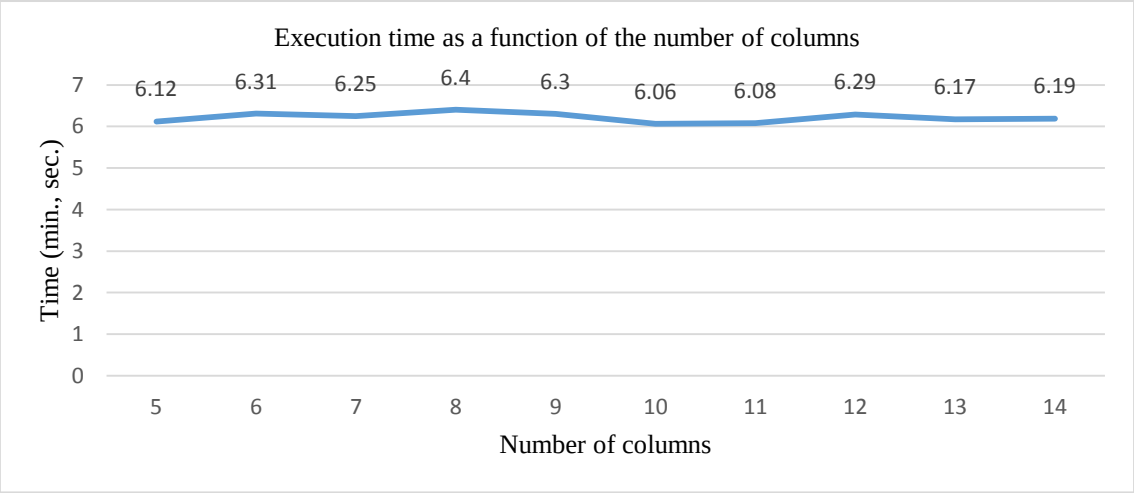


**Figure 7**. Execution times (min, sec) as a function of the number of columns of the data set (5 to 14), with 30,000 rows, 5 equivalence relations, and 1 concept.

The results shown in Figure 8 reflect the execution times achieved by the algorithm as the number of rows of the data set was varied. The variation considered goes from 5,000 to 30,000 rows. It can be seen that the number of rows considered is indeed a determining factor of the execution time, as it is the factor that modifies the complexity of the proposed algorithm the most.

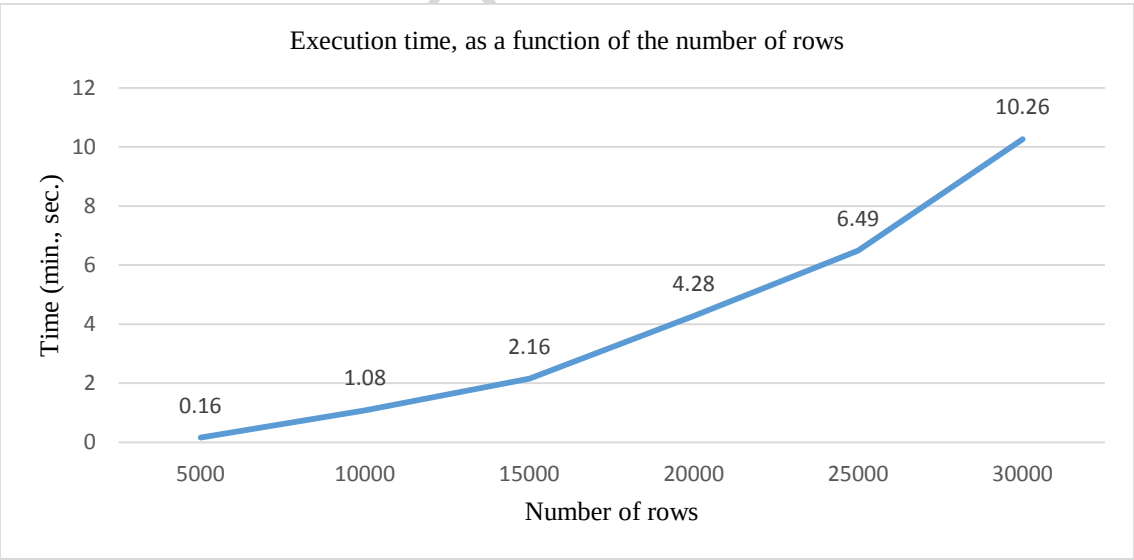


**Figure 8**. Execution time (min., sec.) as a function of the number of elements in the data set, with 14 columns, 5 equivalence relations, and 1 concept.

Figure 9 shows the dependence of execution time on the number of equivalence relations. It was theoretically demonstrated that this dependence is quadratic, and in this plot, we can observe that for small values of *m*, it is in fact almost linear. That is to say, everything indicates that all the constants define a very open parabola; therefore, for small enough values of *m* ($\leq 20$)—which is the most common case— the quasi–linearity of the time complexity of the algorithm with respect to the number of equivalence relations is guaranteed.

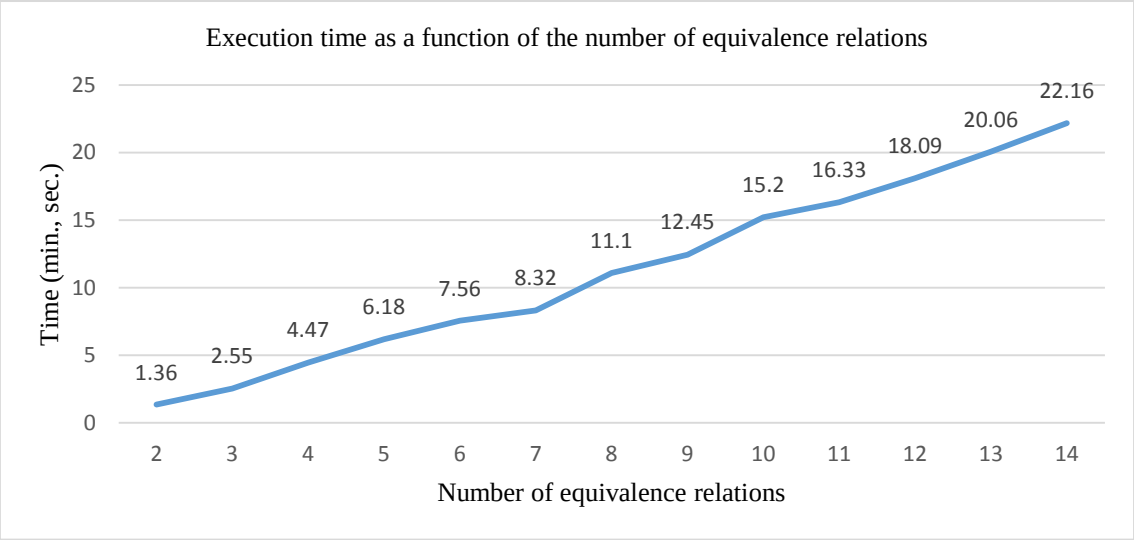


**Figure 9**. Execution time (min., sec.), as a function of the number of equivalence relations, with 30,000 rows and 14 columns in the data set.

### 6.2.- Detection

In the tests for the measurement of the detection quality, the following parameters were chosen:

− The individuals in the table who were the subject of the study were the individuals who satisfied the following CONCEPT: $1 \leq$ people_with_age $\leq 10$.

− The criteria for the analysis were established by the following equivalence relations:

r1: defined from the categorical attribute workclass

c1.1: workclass = [‘private’ OR ‘self-emp-not-inc’ OR ‘self-emp-inc’ OR ‘federal-gov local-gov’ OR ‘state-gov without-pay’]

c1.2: workclass = [‘never-worked’]

r2: defined from the categorical attribute education

c2.1: education = [‘bachelors’ OR ‘some-college’ OR ‘11th’ OR ‘9th’ OR ‘7th-8th’ OR ‘12th’ OR ‘10th’ OR ‘HS-grad’ OR ‘prof-school’ OR ‘assoc-acdm’ OR ‘assoc-voc’ OR ‘masters’ OR ‘doctorate’]

c2.2: education = [‘preschool’ OR ‘1st-4th’ OR ‘5th-6th’]

r3: defined from the categorical attribute marital-status

c3.1: marital-status = [‘married-civ-spouse’ OR ‘divorced’ OR ‘separated’ OR ‘widowed’ OR ‘married-spouse-absent’ OR ‘married-AF-spouse’]

c3.2: marital-status = [‘never-married’]

r4: defined from the categorical attribute occupation

c4.1: occupation = [‘tech-support’ OR ‘craft-repair’ OR ‘other-service’ OR ‘sales’ OR ‘exec-managerial’ OR ‘prof-specialty’ OR ‘handlers-cleaners’ OR ‘machine-op-inspct’ OR ‘adm-clerical’ OR ‘farming-fishing’ OR ‘transport-moving’ OR ‘priv-house-serv’ OR ‘protective-serv’ OR ‘armed-Forces’]

c4.2: occupation = [‘student’]

Note that any element that satisfies the concept and belongs to class c1.1, c2.1, c3.1 or c4.1 is contradicted by the relation $r_x$, keeping in mind that the individuals subject to analysis are children between 1 and 10 years of age who have therefore never worked, have not studied beyond 6th grade, have never married and have the occupation of student.

To verify the correction capabilities of the detection process, the data set was bombarded with a variable set of outliers that were prepared artificially. This test allows us to verify that the outliers are in fact detected. The set of outliers with which the data set was bombarded is shown in Table 2.

**Table 2**. List of outliers introduced to the data set. The fields marked with an asterisk (*) indicate the incorrect data.

| Age | WorkClass | Education | Marital-Status | Occupation |
|---|---|---|---|---|
| 7 | *self-emp-inc | 1st-4th | never-married | student |
| 6 | never-worked | *master | never-married | student |
| 9 | never-worked | *doctorate | never-married | student |
| 9 | never-worked | 5th-6th | never-married | *armed-forces |
| 7 | never-worked | 1st-4th | never-married | *adm-clerical |
| 8 | *self-emp-inc | *master | never-married | student |
| 8 | never-worked | *doctorate | *married-civ-spouse | student |
| 6 | never-worked | 1st-4th | *divorced | *armed-forces |
| 9 | *federal-gov | 5th-6th | never-married | *adm-clerical |
| 3 | *self-emp-inc | *master | *married-civ-spouse | student |
| 7 | never-worked | *doctorate | *divorced | *adm-clerical |
| 2 | *federal-gov | *master | *divorced | *armed-forces |
| 8 | *self-emp-inc | *doctorate | *married-civ-spouse | *armed-forces |

Figure 10 shows the number of outliers detected for different values of the detection threshold μ in the different tests that were performed. The differences among the tests illustrated in Figure 10 are based on the number of outliers in Table 2, which were introduced into the data set.

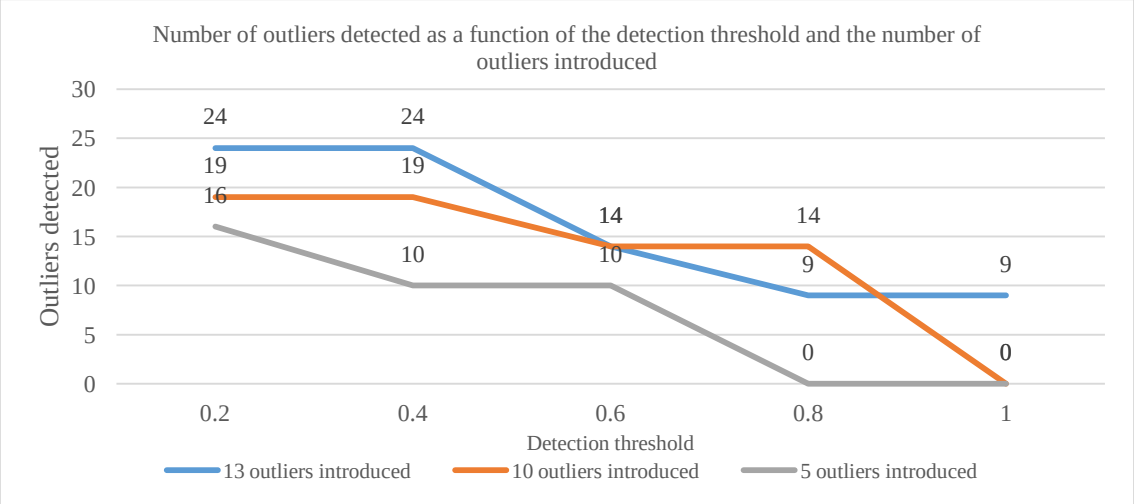


**Figure 10**. Number of outliers detected as a function of the detection threshold μ and the number of outliers introduced.

One aspect of the results that is worth pointing out is that the set of outliers found always contained at least some the outliers introduced. This condition was met both when the number of outliers detected was larger than the number of outliers introduced and when it was smaller. This result indicates that the algorithm correctly detects outliers; some of the outliers detected belong to the set of artificially prepared outlying elements, and the rest were present in the original data set.

Figure 11 shows the number of real outliers in the data set that were detected during the analysis as a function of the number of artificial outliers introduced and the threshold value μ. The larger the value of μ, the smaller the number of outliers detected. This result is to be expected because μ determines the minimum degree of exceptionality necessary, i.e., how many inner boundaries an element needs to be included in and, therefore, how far it is from values considered consistent with the concept being analyzed, for the element to be classified as an outlier.

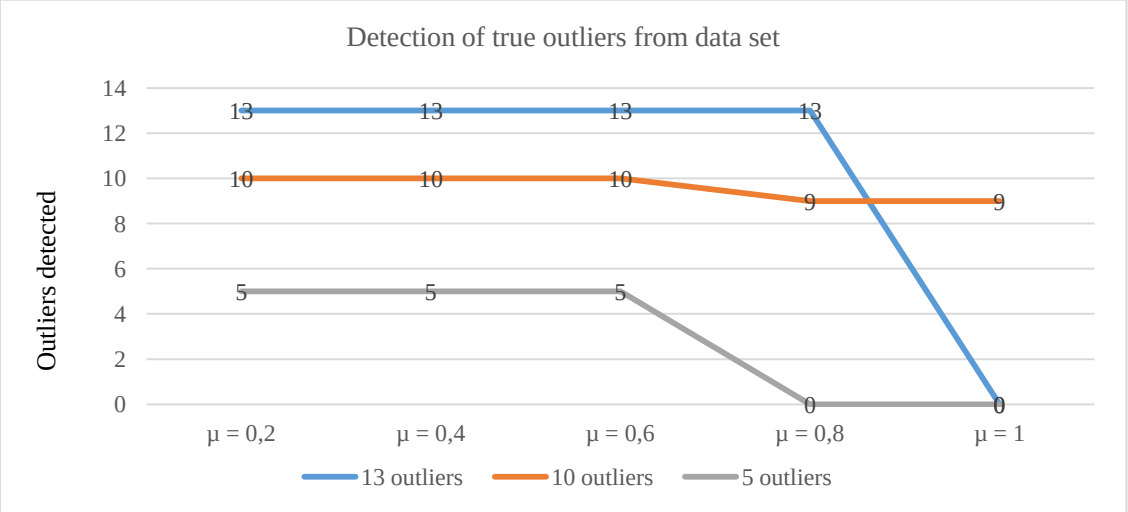


**Figure 11**. Number of real outliers detected in the data set as a function of the number of artificial outliers introduced and the threshold μ.

Figure 12 shows how many of the introduced artificial outliers were detected. Detection of these elements indicates that the algorithm is truly classifying outliers because these elements are outliers by definition—that is, they are by construction in conflict with the concept that is being analyzed.

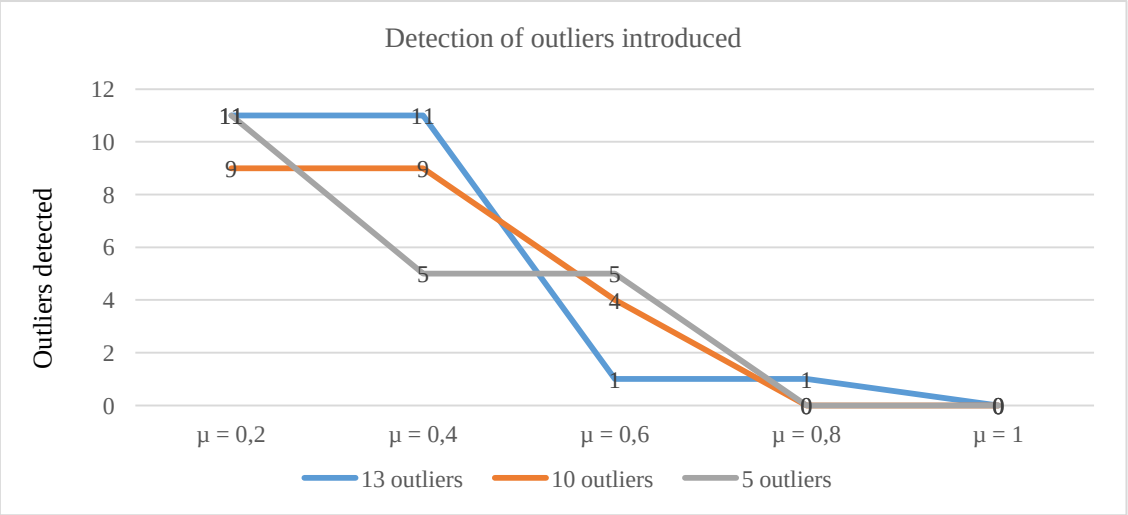


**Figure 12**. Number of artificially introduced outliers detected as a function of the total number of outliers introduced and the threshold μ.

As mentioned above, the elements that are classified as outliers are those that are somehow at odds with the concept that is being analyzed. In the present case, the concept is $1 \leq$ people_with_age $\leq 10$. The variation of the threshold value μ modifies the number of elements that are classified as outliers because a larger threshold indicates that the degree of exceptionality of an element must be larger to be considered an outlier. That is, it must be further from what is considered normal in the set. If an excessively large exceptionality threshold is set, it is possible that no outliers would be detected; this would be an error of the false-negative type. In reality, it is a matter of adjusting the algorithm based on the degree of exceptionality required at each moment.

The above demonstrates the efficiency that can be achieved by the algorithm in terms of detection. The variation of the threshold μ implies a certain fine-tuning of the detection, which in some cases is not achieved. This issue is manifested by the detection getting "stuck", which can be observed in the plots. In some instances, the detection of outliers stops suddenly, which is made evident in the plots by the steep drops to zero. The cause for this lack of refinement is the deterministic nature of the method in terms of classification: because it is based on Rough Set Theory, it inherits that theory's limitations. This issue leads to the supposition that, when a certain degree of declassification is allowed, a better detection quality may be reached in some instances, finally identifying the most contradictory elements as outliers.

## 7.- Conclusions

In this work, a method for the detection of outliers has been proposed with a simple and rigorous theoretical setup, starting from a definition of outliers that is simple, intuitive and computationally viable for large data sets. From this method, an efficient algorithm for outlier mining has been developed, conceptually based on a novel and original approach using Rough Set Theory, which has not been applied in any previous category of classification for the methods of rough set detection.

The proposed algorithm is linear with respect to the cardinality of the data universe over which it is applied, and it is quadratic with respect to the number of equivalence relations used to describe the universe. However, this number of relations merely represents a constant, as it is usually significantly smaller than the cardinality of the universe in question.

In contrast to many other methods that present difficulties in their application depending on the nature of the data to be analyzed, our proposal is applicable to both continuous and discrete

data. The possibility that the data sets may contain a mix of attribute types (e.g., a mix of continuous and categorical attributes) does not present a limitation for the applicability of the proposed algorithm.

Furthermore, the method is applicable to data in table form: the data structure of the Relational Model. The table must be at least in the 1st normal form to guarantee that there are no redundancies, and its attributes must be single–valued. Otherwise, they would contradict with the essence of the method, as it would not be possible to establish equivalence relations from them. All of the above points imply the method's applicability in outlier mining in current large databases.

Therefore, the proposed work is perfect for use in scenarios where cross Decision Suport, Knowledge Discovery and Data Mining, and Big Data. This proposal allows to locate within specified times, elements in a data set that differ from the rest in some degree. This degree can be regulated, so that the elements can be very different or a bit different. This allows: locate errors in data collection to discard or correct, locate elements of a system having a malfunction in comparison with the rest, or detecting a set of records that bear some similarity to each other and differ from the rest revealing a new knowledge. All in datasets where other techniques would be impossible for their temporary costs.

We are currently working on a new version of the algorithm that allows the modeling of uncertain information. In practice, being able to admit a certain level of uncertainty in the classification process may lead to a deeper understanding and to better use of the properties of the data at hand. The theoretical framework developed in this paper is based on the basic model of RS which is unable to model uncertain information. To advance this line, our proposal is to extend again the mathematical framework using concepts and formalisms needed to incorporate uncertainty in the classification of elements. Specifically, there are works on RS modeling using Rough Sets with variable precision [51], and it is necessary to study the feasibility of incorporating this variable precision to the current framework proposed.

**Acknowledgements**

This work was performed as part of the Smart University Project financed by the University of Alicante.

**Appendix**

This appendix provides the formal demonstrations of the expansion of the mathematical framework presented in Section 4, in terms of which the outlier detection algorithm of Section 5 is developed.

Let $U \neq \phi$ be the (finite) universe, and let $r \subseteq UXU$ be an equivalence relation defined over *U*. Let $X \subseteq U$ be a CONCEPT.

**Proposition 8:** $\forall e, e' \subseteq X$, if $e$ is exceptional and $e \subseteq e'$, then $e'$ is also exceptional.

Demonstration: *trivial*

**Lemma 9**: Let $f$ be a *non–redundant exceptional set*, and let *a*: $a \in X \wedge a \notin f$, $\exists f \Leftrightarrow \exists i$, $1 \leq i \leq m$, such that $B_i^c \cup \{a\}$ is an *exceptional set*.

Demonstration:

(⇐)

$B_i^c$ is not *exceptional*, because it does not contain any element that belongs to the *inner boundary* $B_i$ (by virtue of the definition of the complement of a set). Notwithstanding, by hypothesis, if $B_i^c \cup \{a\}$ is an *exceptional set*, then *a* is an *indispensable element* in said set. It would suffice to extract from $B_i^c \cup \{a\}$ all *dispensable elements* and so obtain a set to which *a* belongs, all of whose elements are *indispensable*, making it a *non–redundant exceptional set*.

(⇒ )

By hypothesis, $a \in f$, and *f* is *non–redundant exceptional*. Therefore, *a* is *indispensable* in *f*, which implies that $\exists i,\ 1 \leq i \leq m$, so that $f \cap B_i = \{a\}$ (by virtue of the definition of an *indispensable element*); that is, *a* would be the only representative of the inner boundary of $B_i$ inside *f*. Therefore, $\forall y$ such that $y \in f\text{-}\{a\}$, and it is true that $y \notin B_i$, which in turn implies that $y \in B_i^c$, and so $f\text{-}\{a\} \subseteq B_i^c$.

Let us construct the union of both sets with the set $\{a\}$: $(f\text{-}\{a\}) \cup \{a\} \subseteq B_i^c \cup \{a\}$

$f \subseteq B_i^c \cup \{a\}$.

Because *f* is an *exceptional set*, then $B_i^c \cup \{a\}$ will be *exceptional*, too, by **Proposition 8**.

**Lemma 10:** If $\exists a \in B_i,\ 1 \leq i \leq m$, such that $B_i^c \cup \{a\}$ is not an *exceptional set*, then $\exists\ j,\ 1 \leq j \leq m$, $j \neq i$ such that $B_j \subset B_i$.

Demonstration:

If $B_i^c \cup \{a\}$ is not an *exceptional set*, then there will exist some $j,\ 1 \leq j \leq m$ such that $\forall y \in B_i^c \cup \{a\},\ y \notin B_j$.

Because $a \in B_i$, the absent *inner boundary* cannot be $B_i$. Therefore, $j \neq i$. If $\forall y \in B_i^c \cup \{a\}$, $y \notin B_j$, then $y \in B_j^c$.

It follows that, because $\forall y \in B_i^c \cup \{a\},\ y \in B_j^c$, then,

$$B_i^c \cup \{a\} \subseteq B_j^c \Rightarrow B_i^c \subset B_j^c \Rightarrow B_j \subset B_i$$

Applying the contraposition to **Lemma 10**, the following corollary can be enunciated.

**Corollary 11:** If $\forall j,\ 1 \leq i,\ j \leq m,\ j \neq i$, then it is true that $B_j \not\subset B_i$ (which means that the *inner boundary* $B_i$ does not completely contain any other *inner boundary*). Therefore, $\forall a \in B_i$. $B_i^c \cup \{a\}$ is an *exceptional set*.

**Lemma 12:** Let $a \in X$. If $\exists j,\ j \neq i,\ 1 \leq i,\ j \leq m$ such that $B_j \subset B_i$ and $B_i^c \cup \{a\}$ is an *exceptional set*, then $B_j^c \cup \{a\}$ is also an *exceptional set*.

Demonstration:

$B_j \subset B_i \Rightarrow B_i^c \subset B_j^c \Rightarrow (B_i^c \cup \{a\}) \subset (B_j^c \cup \{a\})$

Therefore, because $B_i^c \cup \{a\}$ is an *exceptional set*, by applying **Proposition 8**, we may conclude that $B_j^c \cup \{a\}$ is also an *exceptional set*.

**Lemma 14:** $\forall i$, $1 \leq i \leq m$, it is true that $E_i \subseteq B_i$.

In other words, all the elements of some particular $E_i$ are elements of the *inner boundary* $B_i$.

Demonstration:

$\forall a \in E_i$, and by the definition of $E_i$, there exists a non–redundant *exceptional set e* such that $e \cap B_i = \{a\}$. Therefore, $a \in B_i$.

**Lemma 15:** Let $a \in X$, $1 \leq i \leq m$, and then $B_i^c \cup \{a\}$ is an *exceptional set* if, and only if, $a \in E_i$.

Demonstration:

($\Rightarrow$)

Because $B_i^c \cup \{a\}$ is an exceptional set in which *a* (see the demonstration of Lemma 9) is an indispensable element (if the element *a* were eliminated from this set, we would be left with the set $B_i^c$, which is known to be non–exceptional, as it does not contain any element of $B_i$), one can obtain a set $f \subseteq B_i^c \cup \{a\}$ such that *f* is a non–redundant exceptional set (eliminating from $B_i^c \cup \{a\}$ all dispensable elements), and in said set, the only representative of the inner boundary $B_i$ is *a* (inferred from the previous arguments); that is, $a \in E_i$.

($\Leftarrow$)

If $a \in E_i$, then there is a *non–redundant exceptional* set *e* that contains the element *a* and $B_i \cap e = \{a\}$.

Making the union of the set $B_i^c$ and the sets that are in both members of the equation, the following is obtained:

$B_i^c \cup (B_i \cap e) = B_i^c \cup \{a\}$

$(B_i^c \cup B_i) \cap (B_i^c \cup e) = B_i^c \cup \{a\}$

$X \cap (B_i^c \cup e) = B_i^c \cup \{a\}$

$B_i^c \cup e = B_i^c \cup \{a\}$

Therefore, because *e* is *exceptional* and $e \subseteq (B_i^c \cup e)$, by virtue of **Proposition 8**, $B_i^c \cup e$ is also *exceptional*. And because

$B_i^c \cup e = B_i^c \cup \{a\}$, then, $B_i^c \cup \{a\}$ is an *exceptional set*.

**References**

[1] L. Akoglu, E. Müller, J. Vreeken eds. ODD '13 Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description. ACM New York (2013).

[2] G. S. David Sam Jayakumar, B.J. Thomas. A New Procedure of Clustering Based on Multivariate Outlier Detection, in: Journal of Data Science 11 (2013), 69-84.

[3] L. Xiong, B. Póczos, J.G. Schneider, A. Connolly, J. VanderPlas. Hierarchical probabilistic models for group anomaly detection. In International Conference on Artificial Intelligence and Statistics (2011), pp. 789-797.

[4] N. Giatrakosa, Y. Kotidisb, A. Deligiannakisc, V. Vassalosb, Y. Theodoridisa. In-network approximate computation of outliers with quality guarantees, in: Information Systems (2013), 38 (8), pp. 1285-1308.

[5] Z. He, X. Xu, J.Z. Huang, S. Deng, (2004). Mining class outlier: concepts, algorithms and applications in CRM, in Expert Systems with Applications 27 (4) (2004), pp. 681-697.

[6]. Raymond. Outlier detection in personalized medicine, in: Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description (ODD '13). (2013) pp. 7-7.

[7] G. Qinglin, W. Kehe, L. Wei, L. Fault Forecast and Diagnosis of Steam Turbine Based on Fuzzy Rough Set Theory, in: Proceedings of the Second International Conference on Innovative Computing, Information and Control - ICICIC'07 (2007) pp. 501-501.

[8] J.A. Cramer, S.S. Shah, T.M. Battaglia, S.N. Banerji, L.A. Obando, K.S. Booksh. Outlier detection in chemical data by fractal analysis, in: Journal of Chemometrics 18 (7-8) (2004), pp 317-326.

[9] E. Achtert, A. Hettab, H.P. Kriegel, E. Schubert, A. Zimek. Spatial Outlier Detection: Data, Algorithms, Visualizations, in: Advances in Spatial and Temporal Databases, LNCS 6849 (2011) pp. 512-516.

[10] Z. Pawlak, Z. Rough sets, in: International Journal of Computer and Information Sciences 11 (5), (1982), pp. 341–356.

[11] Z. Pawlak, Z. Rough Sets: Theoretical Aspects of Reasoning About Data. Kluwer Academic Publishers Norwell, MA, USA, (1992)

[12] F. Gang. Network Teaching Resource Evaluation Method Based on Rough Set Theory, in: Proceedings of the 2009 International Conference on Management of e-Commerce and e-Government, (2009), pp.188-192

[13] H. Liu, W. Kong, T.S. Qiu, G.L. Li. A Neural Network Based on Rough Set (RSNN) for Prediction of Solitary Pulmonary Nodules, in: Proceedings of the International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing, IJCBS'09, (2009) pp.135-138

[14] X. Xiaowen, X. Wei, Z. Beirong, Z. Reconfigurability analysis of manufacturing system based on rough sets, in: Proceedings of the 2nd IEEE International Conference on Computer Science and Information Technology ICCSIT09, (2009) pp.513-517

[15] M. Zhao, H. Liu, A. Abraham, E. Corchado. A Swarm-Based Rough Set Approach for Group Decision Support Systems, in: Proceedings of the Ninth International Conference on Hybrid Intelligent Systems, HIS’09, 3, (2009) pp. 365-369

[16] A. Bouyer, A.H. Abdullah, H. Ebrahimpour, F. Nasrollahi. Fault-Tolerance Scheduling by Using Rough Set Based Multi-checkpointing on Economic Grids, in: 2009 International Conference on Computational Science and Engineering, Vol. 1, (2009) pp.103-109.

[17] P. Yao. Hybrid Classifier Using Neighborhood Rough Set and SVM for Credit Scoring, in: Proceedings of the International Conference on Business Intelligence and Financial Engineering, BIFE'09, (2009), pp.138-142

[18] S. Junding, Ch. Suxia. ROI Extraction Based on Rough Set, in: Proceedings of the International Conference on Environmental Science and Information Application Technology, ESIAT'09, Vol. 3, (2009) pp.207-209

[19] L. Yin, X. Liu, W. Jiang, J. Xie. Evaluation of Enterprise Innovation Ability Based on Rough Set Theory, in: Proceedings of the Asia-Pacific Conference on Information Processing, apcip, Vol. 1, (2009) pp. 471-475

[20] F. Jiang, Y. Sui, C. Cao. Outlier detection using rough sets theory. Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC 2005). Springer, (2005), pp. 79-87.

[21] F. Jiang, Y. Sui, C. Cao. Outlier detection based on rough membership function, in: Rough Sets and Current Trends in Computing, 5th International Conference, RSCTC'06. Kobe, Japan. Springer. (2006) pp. 388-397.

[22] S. Hawkins, H. He, G.J. Williams, R.A. Baxter. Outlier Detection Using Replicator Neural Networks, in: Proceedings Data Warehousing and Knowledge Discovery, DaWaK'02, Springer Berlin Heidelberg, (2002) pp. 170-180

[23] E. Stoimenova, P. Mateev, M. Dobreva, M. Outlier detection as a method for knowledge extraction from digital resources. Review Of The National Center For Digitization-Преглед Нцд, Issn, 109(9), (2006).

[24] C.C. Aggarwal. Towards Exploratory Test Instance Centered Diagnosis in High Dimensional Classification, in: IEEE Transactions on Knowledge and Data Engineering, 19(8), (2007), pp. 1001-1015

[25] F. Angiulli, S. Basta, C. Pizzuti. Distance-Based Detection and Prediction of Outliers, in: IEEE Transactions on Knowledge and Data Engineering, 18 (2), (2006), pp. 145-160

[26] Machine Learning Repository. USA Census Bureau Database. (http://archive.ics.uci.edu/ml/datasets/Census+Income) (02/12/ 2013)

[27] M.E. Otey, A. Ghoting, S. Parthasarathy. Fast Distributed outlier Detection in mixed-attribute data sets. Technical report OSU-CISRC-6/05-TR42, Department of Computer Science and Engineering. The Ohio State University (2005)

[28] M.E. Otey, A. Ghoting, S. Parthasarathy. Fast Lightweight outlier Detection in mixed-attribute data sets. Technical report OSU-CISRC-6/05-TR43, Department of Computer Science and Engineering, The Ohio State University (2005)

[29] S. Shekhar, C. Lu, P. Zhang. A unified Approach to Spatial Outliers Detection, in: GeoInformática, An International Journal on Advances of Computer Science for Geographic Information System, 7(2), (2003), pp. 139-166.

[30] A. Kandel, M. Last. Automated detection of outliers in real-world data, in: Proceedings of the Second International Conference on Intelligent Technologies. Bangkok, Thailand (2001), pp. 292-301.

[31] S. Papadimitriou, H. Kitagawa, P.G. Gibbons, C. Faloutsos. LOCI: Fast Outlier Detection Using the Local Correlation Integral, in: 19th International Conference on Data Engineering, (2003), pp. 315-326.

[32] E. Knorr, R.T.Ng, V. Tucakov. Distance-based outliers: Algorithms and Applications, in: The VLDB Journal, 8 (3-4), (2000), pp. 237-253

[33] V.J. Hodge, J. Austin. A Survey of Outlier Detection Methodologies, in: Artificial Intelligence Review 22(2), (2004), pp. 85-126

[34] E. Knorr, R.T. Ng. Algorithms for mining distance-based outliers in large datasets, in: Proceedings of theVLDB98, (1998), pp. 392-403.

[35] E. Knorr, R.T. Ng. Finding intentional knowledge of distance-based outliers, in: Proceedings of the VLDB99, (2009), pp. 211-222.

[36] M. Schwabacher, S.D. Bay. Mining distance-based outliers in near lineal time with randomization and a simple pruning rule, in: Proc. of 9th annual ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, (2003), pp. 29-38.

[37] J. Tang, Z. Chen, A. Fu, D. Cheung. A Robust Outlier Detection Scheme, in: A robust outlier detection scheme for large data sets. In Proceedings of the 6th Pacific-Asia Conference on Knowledge Discovery and Data Mining (2002), pp. 535-548.

[38] K. Yamanishi, J. Takeuchi. Discovering outlier filtering rules from unlabeled data-combining a supervised learner with an unsupervised learner. In Proceedings of the KDD01, (2001), pp. 389-394.

[39] J. Theodore, K. Ivy, T.N. Raymong. Fast Computation of 2d depth contours, in: ACM SIG KDD, (1998), pp. 224-228.

[40] D. Ren, B. Wang, W. Perrizo. RDF: A density-based Outlier Detection Method using Vertical Data Representation, in: Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM'04), (2004), pp. 503-506

[41] Z. He, S. Deng, X. Xu. Outlier detection integrating semantic knowledge, in: Advances in Web-Age Information Management (2002), pp. 126-131.

[42] M. Petrovsky. A Hybrid Method for Patterns Mining and Outliers Detection in the Web Usage Log, in: In Advances in Web Intelligence, (2003), pp. 318-328.

[43] University of California Irvine. Center for Machine Learning and Intelligent Systems (http://cml.ics.uci.edu) [02/12/2013]

[44] V. Chandola, A. Banerjee V. Kumar. Anomaly detection: A survey, in: ACM Computing Surveys (CSUR), 41(3), (2009), article 15.

[45] P. Sun, S. Chawla. Outlier Detection: Principles, Techniques and Application, in: The 10th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2006), (2006), tutorial.

[46] C. Ching-Hsue, C. You-Shyang, C. Jr-Shian. Classifying Initial Returns of Electronic Firm's IPOs Using Entropy Based Rough Sets in Taiwan Trading Systems, in: Innovative Computing, Information and Control, (2007), pp. 82. IEEE Computer Society

[47] H. Sang Wook, K. Jae-Yearn. Rough Set-based Decision Tree using the Core Attributes Concept, in: The Second International Conference on Innovative Computing, Information and Control 2007, (2007), pp. 298. IEEE Computer Society

[48] M. Hirokane, H. Konishi, A. Miyamoto, F. Nishimura. Extraction of minimal decision algorithm using rough sets and genetic algorithm, in: Systems and Computers in Japan, (2007), 38(4), pp. 39-51.

[49] L. Rokach. An evolutionary algorithm for constructing a decision forest: Combining the classification of disjoints decision trees, in: International Journal of Intelligent Systems, (2008), 23(4), pp. 455-482.

[50] H. Strömbergsson, P. Prusis, H. Midelfart, M. Lapinsh, J. Wikberg, J. Komorowski. Rough set-based proteochemometrics modeling of G-protein-coupled receptor-ligand interactions, in: Proteins: Structure, Function, and Bioinformatics, (2006), 63(1), pp. 24-34.

[51] W. Ziarko. Variable Precision Rough Set Model, in: Journal of Computer and System Sciences, (1993), 46(1), pp. 39-59.

Biographical Note

Francisco Maciá-Pérez was born in Spain in 1968. He received his engineering degree and the Ph.D. degree in Computer Science from the University of Alicante in 1994 and 2001 respectively. He worked as System's Administrator at the University of Alicante form 1996 to 2001. He was an Associate Professor from 1997 to 2001. Since 2001, he is a Professor and currently he is the Vice President for Information Technologies at the University of Alicante. His research interests are in the area of network management, computer networks, smart sensor networks and distributed systems, which are applied to industrial problems. H

Jose Vicente Berna-Martinez was born in Spain in 1978. He received his engineering degree and the Ph.D. degree in Computer Science from the University of Alicante in 2004 and 2011 respectively. From 2006 to 2013, he was an Associate Professor at the University of Alicante, currently he is a Assistant doctor. His research interests are in the area of computer networks, distributed systems, bio-inspired systems and robotics which are applied to industrial problems.

Alberto Fernández Oliva was born in Cuba in 1955. He received his Bachellor and Master degree in in Computer Science from the Havana University in 1979 and 1997 respectively. He received his Ph.D. degree in Computer Science from the University of Alicante in 2010. Full Professor of Computer Science Department at Havana University. His research interests since 2007 are in the area of Data Mining and Knowledge Discovery on Data(outlier detection methods).

Miguel Alfonso Abreu Ortega was born in Cuba in 1987. He is graduated with honors in Computer Science at Havana University. He has been working in subjects relative to Data Mining and Knowledge Data Discovery since 2007. He was a training professor at Havana University.

**Highlights**

- We propose a formal expansion to the theory of rough sets.
- We propose an efficient algorithm for the detection of outliers.
- We have implemented the algorithm and verified the theoretical results.