A New Multiobjective Evolutionary Algorithm for Mining a Reduced Set of Interesting Positive and Negative Quantitative Association Rules

Diana Martín, Alejandro Rosete, Jesús Alcalá-Fdez, *Member, IEEE,* and Francisco Herrera, *Member, IEEE*

***Abstract*—Most of the algorithms for mining quantitative association rules focus on positive dependencies without paying particular attention to negative dependencies. The latter may be worth taking into account, however, as they relate the presence of certain items to the absence of others. The algorithms used to extract such rules usually consider only one evaluation criterion in measuring the quality of generated rules. Recently, some researchers have framed the process of extracting association rules as a multiobjective problem, allowing us to jointly optimize several measures that can present different degrees of trade-off depending on the dataset used. In this paper, we propose MOPNAR, a new multiobjective evolutionary algorithm, in order to mine a reduced set of positive and negative quantitative association rules with low computational cost. To accomplish this, our proposal extends a recent multiobjective evolutionary algorithm based on decomposition to perform an evolutionary learning of the intervals of the attributes and a condition selection for each rule, while introducing an external population and a restarting process to store all the nondominated rules found and to improve the diversity of the rule set obtained. Moreover, this proposal maximizes three objectives—comprehensibility, interestingness, and performance—in order to obtain rules that are interesting, easy to understand, and provide good coverage of the dataset. The effectiveness of the proposed approach is validated over several real-world datasets.**

***Index Terms*—Data mining, MOEA/D-DE, multiobjective evolutionary algorithms, negative association rules, quantitative association rules.**

I. Introduction

IN THE LAST decade, the digital revolution has provided relatively inexpensive and accessible means of collecting and storing data. This unlimited growth of data has led to a situation in which the knowledge extraction process is more difficult and, in most cases, leads to problems of scalability and/or complexity [1]. Association discovery is one of the most common data mining (DM) techniques used to extract interesting knowledge from large datasets [2]. Association rules are used to identify and represent dependencies between items in a dataset [3]. These are representations of the type $X \rightarrow Y$, in which X and Y are item sets and $X \cap Y = \oslash$. Therefore, if the items in X exist in an example then it is highly probable that the items in Y are also in the example, and X and Y should not have items in common [4], [5]. A high number of previous studies on mining association rules have focused on datasets with discrete or binary values; however, in real-world applications, data usually consists of quantitative values. Because of this, different studies have been presented for mining quantitative association rules (QARs) from datasets with quantitative values [6], [7].

Most of these algorithms usually extract positive QARs without paying particular attention to negative QARs. Nevertheless, rules such as $X \rightarrow \neg Y$ may be worth taking into account, as they relate the presence of X to the absence of Y [8]. Negative association rules consider the same sets of items as positive association rules but, in addition, may also include negated items within the antecedent ($\neg X \rightarrow Y$) or the consequent ($X \rightarrow \neg Y$) or both of them ($\neg X \rightarrow \neg Y$). In recent years, some researchers have proposed methods for mining positive and negative association rules from quantitative data [9]–[12]. The researchers deal with two key problems in negative association rule mining: how to effectively search for interesting itemsets and how to effectively identify interesting negative association rules.

Many evolutionary algorithms (EAs) [13], have been proposed in the literature for extracting a set of QARs from datasets [14]–[16]. EAs, particularly genetic algorithms (GAs) [17], are considered to be one of the most successful search techniques for complex problems and have proved to be an important technique for learning and knowledge extraction. These algorithms usually consider only one evaluation criterion in measuring the quality of the generated rules. Recently, some researchers have framed the extraction of association rules as a multiobjective (rather than a single objective) problem, taking into account several objectives in the process of extracting association rules [18], [19]. This approach removes some of the limitations of the mono-objective algorithms and allows us to jointly optimize

Manuscript received November 13, 2012; revised April 8, 2013 and July 26, 2013; accepted October 2, 2013. Date of publication October 9, 2013; date of current version January 27, 2014. This work was supported in part by the Spanish Ministry of Education and Science under Grant TIN2011-28488, in part by the CEI program of the Spanish Ministry of Science and Innovation under Grant PYR-2012-13 CEI BioTIC GENIL (CEB09-0010), and in part by the Andalusian Government under Grant P10-TIC-6858. This paper was recommended by Associate Editor C. A. Coello Coello.

D. Martín and A. Rosete are with the Department of Artificial Intelligence and Infrastructure of Informatics Systems, Higher Polytechnic Institute J.A Echeverría, La Habana 19390, Cuba (e-mail: dmartin@ceis.cujae.edu.cu; rosete@ceis.cujae.edu.cu).

J. Alcalá-Fdez and F. Herrera are with the Department of Computer Science and Artificial Intelligence, CITIC-UGR, University of Granada, Granada 18071, Spain. F. Herrera is also with the Faculty of Computing and Information Technology - North Jeddah, King Abdulaziz University, 21589, Jeddah, Saudi Arabia (e-mail: jalcala@decsai.ugr.es; herrera@decsai.ugr.es).

Digital Object Identifier 10.1109/TEVC.2013.2285016

1089-778X © 2013 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

several measures in order to mine a set of rules that are interesting, easy to understand, and with good coverage of the dataset.

Multiobjective evolutionary algorithms (MOEAs) [20], [21] provide an interesting method with which to approach problems of a multiobjective nature, as they generate a family of equally valid solutions, in which each solution tends to satisfy a criterion to a greater extent than another. For this reason, some MOEAs have been applied to mine QARs (by considering several measures as objectives) [22], [23] where each solution in the Pareto front represents a QAR with different degree of tradeoff between the different measures.

Recent MOEAs are based on decomposition (MOEA/D [24] and MOEA/D-DE [25]), which explicitly decomposes the multiobjective optimization problem into N scalar optimization subproblems, and also optimizes them simultaneously. These approaches have shown some advantages over other MOEAs, presenting lower computational complexity and a better performance in three-objective continuous test instances. Note MOEA/D [24] won the CEC2009 competition. These reasons have given rise to a growing interest in these approaches within the MOEA research community.

In this paper, we propose MOPNAR, a new MOEA, in order to mine with a low computational cost a reduced set of positive and negative QARs (PNQARs) that are interesting, easy to understand, and with a good trade-off between the number of rules, support, and coverage of the dataset. To accomplish this, our proposal extends the recent MOEA based on decomposition MOEA/D-DE [25] in order to perform a condition selection and an evolutionary learning of the intervals of the attributes for each rule, maximizing three objectives: comprehensibility, interestingness, and performance. Moreover, this proposal introduces a restarting process and an external population (EP) to the evolutionary model in order to promote diversity in the population, store all the nondominated rules found, and improve the coverage of the datasets.

In order to assess the performance of the proposed approach, we have presented an experimental study using nine real-world datasets, with a number of variables ranging from 4 to 91 and a number of examples ranging from 40 to 22 784. We have developed the following studies. First, we have analyzed the performance of our method with another evolutionary approach for mining PNQARs proposed by Alatas *et al.* [9] (which will be called Alatasetal in this paper). Second, we have compared the performance of our approach with three mono-objective evolutionary approaches (GENAR [14], EARMGA [15] and GAR [26]) and three MOEAs (MODENAR [18], MOEA_Ghosh [19], and ARMMGA [23]) for mining QARs. Third, we have compared the results obtained from the comparison with two other classical approaches for mining association rules (Apriori [6], [27], and Eclat [28]) and another classical MOEA (NSGA-II [29]). Fourth, we have studied the scalability of the proposed approach. Finally, we analyze some of the rules obtained by our proposal.

This paper is arranged as follows. Section II introduces some basic definitions of PNQARs and some quality measures. Section III details the evolutionary learning components proposed to mine a reduced set of high quality PNQARs. Section IV shows and discusses the results that are obtained with nine real-world datasets. Finally, in Section V, some concluding remarks are made.

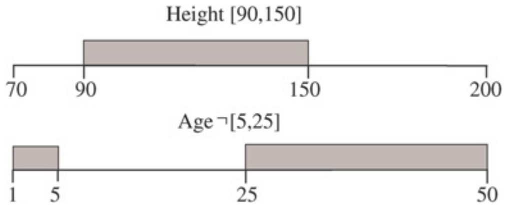


Fig. 1. Example of a positive and negative item.

II. Preliminary: Positive and Negative Quantitative Association Rules

Many previous studies for mining association rules have focused on datasets with binary or discrete values; however, the data in real-world applications usually consists of quantitative values. The association rules obtained from datasets with quantitative values are known as QARs [6], where each item is a pair attribute-interval. For instance, a positive QAR could be $Age \in [30, 52]$ and $Salary \in [3000, 3500] \rightarrow NumCars \in [3, 4]$. The use of QARs to solver real-world problems is a widespread practice in a wide range of sciences, such as biology [30], health [31], etc. The classical algorithms can only be used directly in the discovery of positive QARs with difficulty, because the numerical attributes typically contain many distinct values. A commonly used method is to partition the domains, introducing new attributes with intervals. Thus, the support for a crisp value is likely to be low, while the support for intervals is much higher. However, the given intervals may have a critical influence on the final mining results and the task partition is a critical problem in the extraction of QARs because the information is not classified. For this reason, some approaches have also introduced a learning of the intervals to handle continuous domains in the extraction of QARs [9], [14], [15], [18], [19], [23], [26].

Most of these algorithms have only focused on positive rules, i.e., only those itemsets appearing frequently together will be discovered. However, the negative association rules may also be interesting as they offer information that could be used to support decisions for applications. Negative association rules [8] consider the same sets of items as positive association rules but may also include negated items within the antecedent ($\neg X \rightarrow Y$) or the consequent ($X \rightarrow \neg Y$) or both ($\neg X \rightarrow \neg Y$). For instance, a simple example of a negative QAR is Weight ∈ ¬ [10,25] and Height ∈ [90, 150] → Age ∈ ¬ [5, 25].

Notice that, positive association rules only include positive items whereas negative association rules include at least one negative item. Fig. 1 shows the domain of the positive item $Height \in [90, 150]$ and the negative item $Age \in \neg[5, 25]$.

Support and confidence are the most common measures used to assess QARs, both of them based on the support of an itemset. The support of the itemset I is defined as

$$SUP(I) = \frac{|\{e \in D \mid I \in e\}|}{|D|} \tag{1}$$

where the numerator is the number of examples in the dataset D covered by the itemset I, and $\mid D \mid$ is the number of examples in the dataset. Thus, the support and confidence for a rule $X \rightarrow Y$ are defined as

$$support(X \rightarrow Y) = SUP(XY) \tag{2}$$

$$confidence(X \rightarrow Y) = \frac{SUP(XY)}{SUP(X)}. \tag{3}$$

The classic techniques for mining association rules attempt to discover rules whose support and confidence are greater than the user-defined threshold's minimum support (minSup) and minimum confidence (minConf). However, several authors have noted some drawbacks of this framework that lead it to find many misleading rules [8], [32], [33]. On one hand, the confidence measure does not detect statistical independence or negative dependence between items, because it does not take into account the consequent support. On the other hand, itemsets with very high support are a source of misleading rules because they exist in most of the examples and therefore any itemset may seem to be a good predictor of the presence of the high-support itemset.

In recent years, several researchers have proposed other measures for the selection and ranking of examples according to their potential interest to the user [34], [35]. We briefly describe some of those that have been used in this paper.

The conviction [33] measure analyzes the dependence between X and $\neg Y$, where $\neg Y$ means the absence of Y. Its domain is [0,∞), where values less than one represent negative dependence, a value of one represents independence, and values higher than one represent positive dependence. The main drawbacks of this measure are that it is difficult to define a conviction threshold because its range is not bounded, and this measure does not decrease when the support of the antecedent increases and the rest of the parameters remain the same. Conviction for a rule $X \rightarrow Y$ is defined as

$$conviction(X \rightarrow Y) = \frac{SUP(X)SUP(\neg Y)}{SUP(X \neg Y)}. \tag{4}$$

Notice that this measure obtains an undefined value (NAN) when SUP(Y)=1. In this case, we will consider the conviction value to be one, because it denotes independence. The lift [36] measure represents the ratio between the confidence of the rule and the expected confidence of the rule. As with conviction, its domain is [0,∞), where values less than one imply negative dependence, one implies independence, and values higher than one imply positive dependence. The main drawback of this measure is that it is difficult to define a lift threshold because its range is not bounded. Lift for a rule $X \rightarrow Y$ is defined as

$$lift(X \rightarrow Y) = \frac{SUP(XY)}{SUP(X)SUP(Y)}. \tag{5}$$

The certainty factor (CF) [37] is interpreted as a measure of variation of the probability that Y is in a transaction when we consider only those transactions where X is present. Its domain is [-1,1], where values less than zero represent negative dependence, zero represents independence, and values higher than zero represent positive dependence. This measure for a rule $X \rightarrow Y$ is defined in three ways depending on whether the confidence is less than, greater or equal to $SUP(Y)$

$if\ confidence(X \rightarrow Y) > SUP(Y)$

$$\frac{confidence(X \rightarrow Y) - SUP(Y)}{1 - SUP(Y)} \tag{6}$$

$if\ confidence(X \rightarrow Y) < SUP(Y)$

$$\frac{confidence(X \rightarrow Y) - SUP(Y)}{SUP(Y)} \tag{7}$$

$Otherwise\ is$ 0.

The netconf [38] measure evaluates the rule based on the support of the rule and its antecedent and consequent support. Netconf obtains values in [-1,1], where positive values represent positive dependence, negative values represent negative dependence, and zero represents independence. Netconf for a rule $X \rightarrow Y$ is defined as

$$netconf(X \rightarrow Y) = \frac{SUP(XY) - SUP(X)SUP(Y)}{SUP(X)(1 - SUP(X))}. \tag{8}$$

Notice that if this measure obtains NAN we will consider the nefconf value to be zero, because it denotes independence. Finally, the yule'sQ [39] measure represents the correlation between two possibly related dichotomous events. This measure takes on values in [-1,1] where one implies a perfect positive correlation, -1 implies a perfect negative correlation, and zero implies that there is no correlation. This measure satisfies almost all the properties for interesting measures [34], [35] that have been proposed in the literature. Notice that as netconf if this measure obtains NAN we will consider there to be no correlation. Yule's Q for a rule $X \rightarrow Y$ is defined as

$$\frac{SUP(XY)SUP(\neg X \neg Y) - SUP(X \neg Y)SUP(\neg XY)}{SUP(XY)SUP(\neg X \neg Y) + SUP(X \neg Y)SUP(\neg XY)}. \tag{9}$$

III. New Multiobjective Evolutionary Algorithm for Mining Positive and Negative Quantitative Association Rules: MOPNAR

This section describes our proposal for obtaining a reduced set of PNQARs with a good trade-off between the number of rules, support and coverage, considering three objectives: comprehensibility, interestingness, and performance. This proposal extends the MOEA/D-DE algorithm [25] in order to perform an evolutionary learning of the rules and introduces two new components to its evolutionary model: an EP and a restarting process. In the following, we will explain in detail all their characteristics (see Section III-A–III-E) and present a flowchart of the algorithm (see Section III-F).

A. EP and restarting process within MOEA/D-DE

We extend the MOEA based on decomposition MOEA/D-DE [25], which decomposes the multiobjective optimization problem into N scalar optimization subproblems and uses an EA to optimize these subproblems simultaneously. In order to store all the nondominated rules found, provoke diversity in the population, and improve the coverage of the datasets, we have introduced an EP and a restarting process to the evolutionary model of this MOEA. The EP will keep all the nondominated rules found and will be updated with the newly generated offspring for each solution of the population. The redundant nondominated rules will be removed from EP in order to avoid the overlapping rules. A rule is considered redundant if the intervals of all its variables are contained within the intervals of the variables of another rule. The size of the EP is not limited, which allows us to:

1) obtain a larger number of rules of the Pareto front regardless of the size of the population;
2) reduce the size of the population, following a dataset-independent approach.

However, the EP will usually contain a reduced set of rules because the non-dominance criteria allows us to maintain only the rules of the Pareto front and that the redundant rules are removed.

To move away from local optima and provoke diversity in the population, the restarting process will be applied when the number of new individuals of the population in one generation is less than $\alpha\%$ of the size of the current population (with α determined by the user, usually at 5%). In this case, the examples covered by the rules in the EP are marked and the process of initialization of the population is again applied in order to restart the population from examples uncovered by the rules in the EP (see Section III-C). Moreover, the EP will be updated with the new population following the non-dominance criteria and the redundant rules will be removed. This process allows us to perform a good exploration of the search spaces and to improve the coverage of the dataset.

B. Objectives

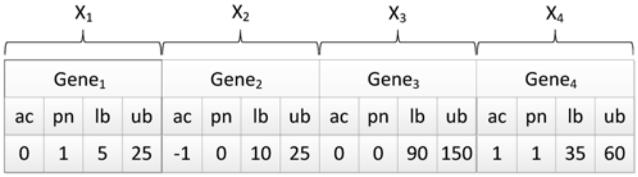
Three objectives are maximized for this problem: interestingness, comprehensibility, and performance. Performance represents the attempt to improve the coverage of the dataset in order to extract more interesting knowledge from it. Performance is the product of support and CF (see Section II), which allows us to mine a set of accurate rules with a good trade-off between local and general rules.

We are interested only in very strong rules [32], which indicate a strong dependence between items and avoid the problem of high-support itemsets (see Section II). Notice that negative association rules allow us to represent negative dependence, thus we are interested in rules that have CF > 0. Thus, a rule $X \rightarrow Y$ must satisfy:

1) $CF(X \rightarrow Y) > 0$;
2) $support(X \rightarrow Y) >$ minSup;
3) $\neg(support(X \rightarrow Y) >$ (1 - minSup))

where minSup is a minimum coverage of the dataset that the rules have to fulfill; we use zero for this value. This measure can obtain values in the interval [0, 1]. A rule with a performance value near to one may be more useful to the user.

Interestingness is a means by which we can measure how interesting the rule may be, allowing us to extract only those rules that may be of interest to the users. Here, we have used the well-known interestingness measure lift (see Section II). This measure can detect negative dependence, independence or positive dependence between items and its range is not bounded, allowing us to better denote the difference between the rules for this objective and to reduce the number of draws.



X_1				X_2				X_3				X_4			
Gene$_1$				Gene$_2$				Gene$_3$				Gene$_4$			
ac	pn	lb	ub	ac	pn	lb	ub	ac	pn	lb	ub	ac	pn	lb	ub
0	1	5	25	-1	0	10	25	0	0	90	150	1	1	35	60

Fig. 2. Example of a chromosome.

Finally, comprehensibility tries to quantify how easy it is to understand the rule [40]. The rules generated may involve a large number of attributes, making them difficult to understand. The user will be highly unlikely to use the rules generated if they are not understood. For the purposes of this paper, we measure the comprehensibility of a rule $X \rightarrow Y$ according to the number of attributes it contains. This is defined as

$$comprehensibility(X \rightarrow Y) = 1/Attr_{X \rightarrow Y} \tag{10}$$

where $Attr_{X \rightarrow Y}$ is the number of attributes involved in the rule.

C. Coding scheme and initial gene pool

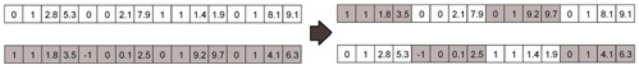
A chromosome is a gene vector, representing the attributes and intervals of the rule. For the purpose of this paper, we use a positional encoding in which the i-th attribute is encoded in the i-th gene used. Notice that the same attribute cannot appear more than once in a rule. Each gene consists of four parts, described below, in order to combine the learning of the intervals with the condition selection.

1) *ac* indicates whether a gene is involved in the rule. If this part is 1 or 0, this attribute is part of the consequent or antecedent of the rule, respectively, whereas if it is -1, this attribute is not involved in the rule. The genes that have 1 in their first parts will make up the consequent of the rule, while genes that have 0 will make up the antecedent.
2) *pn* indicates whether an interval is positive or negative. When this part is 1, the interval is positive and when this part is 0 the interval is negated in the rule.
3) *lb* represents the lower bound of the interval of the attribute.
4) *ub* represents the upper bound of the interval of the attribute.

Notice that if the attribute is nominal, *lb* and *ub* will be equal, representing only one value of the nominal attribute. Thus, a chromosome C_T is coded in the following way, where n is the number of attributes in the dataset

$$\begin{aligned}&Gene_i = (ac_i, pn_i, lb_i, ub_i), \quad i = 1, \ldots, n\\&C_T = Gene_1 Gene_2 \ldots Gene_n.\end{aligned}$$

For instance, let us consider a simple dataset with four attributes X_1, X_2, X_3 and X_4. Let us suppose that we select at random the attributes X_1 and X_3 for the antecedent and X_4 for the consequent of the rule. Based on this definition, Fig. 2 shows the chromosome, which represents the rule $X_1 \in [5, 25]$ and $X_3 \in \neg[90, 150] \rightarrow X_4 \in [35, 60]$.



0	1	2.8	5.3	0	0	2.1	7.9	1	1	1.4	1.9	0	1	8.1	9.1
1	1	1.8	3.5	-1	0	0.1	2.5	0	1	9.2	9.7	0	1	4.1	6.3

1	1	1.8	3.5	0	0	2.1	7.9	0	1	9.2	9.7	0	1	8.1	9.1
0	1	2.8	5.3	-1	0	0.1	2.5	1	1	1.4	1.9	0	1	4.1	6.3

Fig. 3. Simple example of the crossover operator.

To prevent the intervals from increasing to such an extent that they cover the entirety of the domain, we have defined amplitude. This is the maximum size that the interval of a given attribute can attain. The amplitude of an attribute i is therefore defined as

$$amplitude_i = (Max_i - Min_i)/\gamma \tag{11}$$

where γ is a value given by the system expert that determines the tradeoff between the generalization and specificity of the rules, and Min_i and Max_i are the minimum and maximum values of the domain of attribute i, respectively. Notice that if the interval is negative, amplitude represents the minimum size that the interval of an attribute can attain.

The initial population will be composed of a set of rules containing only one attribute in the consequent and a good coverage of the dataset. In order to do this, we first make a random selection of the attributes that will form part of the antecedent and consequent of the rule. An example is then selected at random and the interval of each attribute that has a size equal to 50% of the amplitude of each attribute is generated. The values of the selected example are placed in the center of each interval. If some bound of the intervals exceeds the domain of the attribute this will be replaced by the bound of the domain. After that, we select at random, whether the intervals will be positive or negative. Finally, the examples covered by this rule are marked from the dataset. This process is repeated for unmarked examples until the initial population is completed. Notice that, if all the examples are marked and the initial population is not completed, then all the examples will be unmarked again and the process will be repeated until the initial population is completed. The EP is initialized with the nondominated rules of the initial population.

D. Genetic operators

By interchanging the genes of the parents at random, the crossover operator generates two offspring (exploration). An example of how this operator works is given in Fig. 3.

The mutation operator consists of modifying, at random, the four parts (ac, pn, lb and ub) of a gene selected randomly. This operator selects at random one of the bounds of the interval and increases or decreases its value randomly. We have to be particularly careful not to surpass the fixed value of amplitude. In that sense, the way that we modify the interval is similar to that calculated in the initialization process. The value for ac and pn is randomly selected within the set {-1,0,1} and {0,1} respectively.

The repairing operator is used to change those rules that either have more than one attribute in the consequent or do not have an antecedent or consequent. If the consequent contains more than one attribute, one of them is selected at random to be the consequent and the remaining attributes are passed to the antecedent. If there is no attribute in the antecedent and/or consequent these are selected at random from the attributes that are not involved.

Finally, the repairing operator decreases the sizes of the intervals until the number of examples covered is smaller than that covered by the original intervals, in order to obtain simpler rules. Alternatively, if the interval is negated then the interval is increased, reducing the domain that it covers.

Notice that we have used common genetic operators that work well for mining PNQARs instead of the genetic operators of MOEA/D-DE for multiobjective optimization problems.

E. Evolutionary multiobjective model MOEA/D-DE

With the previous modifications, the evolutionary model will be as follows. First, the evolutionary model of MOEA/D-DE generates a weight vector (λ) for each subproblem, which are used to calculate the value of each subproblem for the decomposition approach (g). Then, a neighborhood (B) is selected for each weight vector, where the T closest weight vectors to a weight vector represent its neighborhood. Then, the algorithm generates an initial population, initializes the reference point (z) with the best values found so far for each objective, and initializes the EP with the nondominated rules of the initial population. Then, two offspring are generated by crossover, mutation and repairing from a solution of the population and another is selected at random from its neighborhood or from the population with a δ probability (δ is defined by the user). These offspring are used to update the reference points and replace some of the solutions of the current population with worse values for the decomposition approach. Notice that the maximal number of solutions replaced by an offspring solution in MOEA/D-DE is bound by η_r, which should be set to be much smaller than T. These steps are repeated for each solution in the population and then EP is updated. Finally, the restarting process is applied at the end of each generation when the number of new subproblems in the population is less than α%. This process is iterated until a stopping condition is satisfied (for more information, see [25]).

There are several approaches to converting the multiobjective optimization problem into a number of scalar optimization problems. In [24], the authors presented three different decomposition approaches and recommended the use of the Tchebycheff approach [41]. In this paper we have also used the Tchebycheff approach, which minimizes the distance between the objective values of the solutions and the reference points.

F. Flowchart of the algorithm

According to the above description, the proposed algorithm for mining PNQARs can be summarized in the following flowchart.

Input: 1) N population size;
2) $nTrials$ number of evaluations;
3) m number of objectives;
4) P_{mut} probability of mutation;
5) $\lambda^1, \ldots \lambda^N$ a set of N weight vectors;

```
    6) $T$ the number of weight vectors in the neighborhood of each weight vector;
    7) $\delta$ the probability that parent solutions are selected from the neighborhood;
    8) $\eta_r$ the maximal number of solutions replaced by each child solution;
    9) $\gamma$ factor of amplitude for each attribute of the dataset;
    10) $\alpha$ difference threshold.
  Output: EP
Step 1: Initialize.
    a) Compute the Euclidean distances between any two weight vectors and then work out the $T$ closest weight vectors to each weight vector. For each $i = 1, ..., N$ set $B_i = \{i_i, ...$
       $i_T\}$ where $\lambda^{i_1}, ...\lambda^{i_T}$ are the $T$ closest weight vectors to $\lambda^i$.
    b) Generate the initial population with $N$ chromosomes.
    c) Evaluate the initial population.
    d) Initialize $z = (z_1, ..., z_m)$ by setting $z_j = \max_{1 \leq i \leq N} f_j(x^i)$, $j = 1, ..., m$.
    e) Initialize the EP.
Step 2: Update. For each $i = 1, ..., N$ do the following.
    a) Uniformly randomly generate a number $rand$ from [0,1]. Then set
       $$P = \begin{cases} \text{B(i)} & \text{if } rand < \delta \\ \{1, ..., N\} & \text{otherwise.} \end{cases}$$
    b) Set $r_1 = i$ and randomly select $r_2$ from $P$. The solutions $x^{r_1}$ and $x^{r_2}$ are crossed, generating two offspring g: $y_1$ and $y_2$. Next, the mutation and repairing operators are applied for the two offspring.
    c) Evaluate the new individuals. For each $y_k$, $k \in \{1, 2\}$.
        i) Update of $z$: For each $j = 1, ...m$, if $z_j < f_j(y_k)$, then set $z_j = f_j(y_k)$.
        ii) Update solutions: Set $c = 0$ and then do the following.
            A) If $c == \eta_r$ or $P$ is empty go to Step 3. Otherwise randomly pick an index $l$ from $P$.
            B) If $g(y_k|\lambda^l, z) \leq g(x^l|\lambda^l, z)$, then $x^l = y_k$ and $c = c + 1$
            C) Remove $l$ from $P$ and go to a).
Step 3: Update of EP: remove from EP all the vectors dominated by $i$, $i = 1, ..., N$, then add $i$ to EP if no vectors in EP dominate it.
Step 4: Remove redundance from the EP.
Step 5: If the difference between the current population and previous population is less than $\alpha\%$, restart the population.
Step 6: If the maximum number of evaluations is not reached, go to Step 2.
Step 7: Remove redundance from the EP.
Step 8: The EP is returned.
```

IV. Experimental analysis

Several experiments have been carried out in this paper to analyze the[1] performance of our proposal. In order to present them, this[2] section is organized as follows.

1) In Section IV-A, we describe the real-world datasets that are used in these experiments.
2) In Section IV-B, we introduce a brief description of the algorithms considered for comparison and we show the configuration of the algorithms (determining all the parameters used).
3) In Section IV-C, we compare our approach with the algorithm Alatasetal [9] because this evolutionary algorithm can mine the PNQARs.
4) In Section IV-D, we compare the performance of our approach with three mono-objective evolutionary approaches (EARMGA [15], GAR [26] and GENAR [14]) and three MOEAs (ARMMGA [23], MODENAR [18] and MOEA_Ghosh [19]) for mining positive QARs.
5) In Section IV-E, we compare our approach with two classical positive association rules extraction algorithms (Apriori [6], [27] and Eclat [28]) and another classical MOEA (NSGA-II [29]).
6) In Section IV-F, we analyze the scalability of the proposed approach.
7) In Section IV-G, we study some of the rules obtained by our proposal.

A. Datasets

In order to analyze the performance of the proposed approach, we have considered nine real-world datasets. Table I summarizes the main characteristics of the nine datasets, which are available in the repository KEEL-dataset [42] from which they can be downloaded (Available at), where Attributes(R/I/N) is the number of (Real/Integer/Nominal) attributes in the data and Examples is the number of examples. To develop the different experiments, we consider the average results of five runs for each dataset.

B. Algorithms Considered for Comparison and Set Up

In these experiments, we compare the proposed approach with ten other algorithms, which are available from the KEEL software tool [43]. A brief description of these algorithms follows.

1) Genetic algorithm for automated mining of both positive and negative quantitative association rules (Alatasetal) [9]: This algorithm designs a GA to simultaneously search for intervals of quantitative attributes and to discover the positive and negative QARs that these intervals

[1] This database was designed on the basis of data provided by U.S. Census Bureau [http://www.census.gov] (under Lookup Access [http://www.census.gov/cdrom/lookup]: Summary Tape File 1).

[2] This study was performed at the 2nd Department of Medicine, 1st Faculty of Medicine, Charles University and Charles University Hospital, under the supervision of Prof. F. Boudk with the collaboration of M. Tomeckov and Assoc. Prof. J. Bultas. The data were transferred to electronic form by the European Centre of Medical Informatics, Statistics and Epidemiology, Charles University and Academy of Sciences. The data resource is on the web page http://euromise.vse.cz/challenge2004. At present, the data analysis is supported by a grant from the Ministry of Education CR Nr LN 00B 107.

TABLE I

DATASETS CONSIDERED FOR THE EXPERIMENTAL STUDY

Names	*Attributes(R/I/N)*	*Examples*
Bolts (bol)	8 (2/6/0)	40
S. Flare (fla)	12 (0/0/12)	1066
House_16H (hh)[1]	17 (10/7/0)	22784
Movement Libras (mov)	91 (90/0/1)	360
Pollution (pol)	16 (16/0/0)	60
Quake (qua)	4 (3/1/0)	2178
Segment (seg)	20 (19/1/0)	2310
Stock Price (sto)	10 (10/0/0)	950
Stulong (stu)[2]	5 (5/0/0)	1419

conform to in a single run. The chromosomes represent rules, in which each gene has four parts. The first part represents the antecedent or consequent of the rule, the second part represents the positive or negative ARs, and the third and fourth part represent the lower and upper bound of the item interval, respectively. The proposed GA performs a dataset-independent approach that does not rely upon the minSup and minConf thresholds.

2) Evolutionary association rules mining with genetic algorithm (EARMGA) [15]: This algorithm uses a GA, which does not require a user-specified threshold for minSup, to identify QARs. Each chromosome encodes a generalized k-rule, where k indicates the desired length. The most interesting rules are returned according to the interestingness measure defined by the fitness function, which is based on the support of the rule and its antecedent and consequent support.
3) Genetic association rules (GENAR) [14]: This algorithm mines association rules in numeric datasets by using a GA. Each chromosome encodes an association rule, containing maximum and minimum intervals of each numeric attribute. The length of the rules is always fixed to the number of attributes, only the last attribute forms the consequent. The objective function considers the number of records covered by the rule and penalises those which have already covered the same records in the dataset.
4) Genetic association rules (GAR) [26]: This algorithm is an extension of GENAR [14], which searches for frequent itemsets in numeric datasets without needing to discretize the attributes. Each chromosome is a k-itemset, in which each gene represents the maximum and minimum values of the attributes that belong to the k-itemset. This algorithm finds frequent itemsets, and it is therefore necessary to run another procedure afterwards in order to generate association rules.
5) Multiobjective association rules with genetic algorithms (ARMMGA) [23]: This algorithm is an MOEA based on the EARMGA algorithm for mining QARs without taking the minSup and minConf into account. According to the comments of the authors, the most important aspect of this algorithm is that its fitness function only specifies the order of chromosomes in the population and does not have any other effect on the GA operator, using this order as a selection criterion. The population with the best average fitness, which is based on the product of the support and confidence of the rules, will be returned.
6) Multiobjective differential evolution algorithm for mining numeric association rules (MODENAR) [18]: This algorithm uses a multiobjective differential evolution algorithm based on Alatasetal [9] to mine accurate and comprehensible QARs without specifying minSup and minConf. This algorithm uses the same coding scheme for the chromosomes as Alatasetal but without the second part. MODENAR weighs four objectives to improve the quality objectives of the rules: support, confidence, comprehensibility, and amplitude of the domain of the attributes.
7) Multiobjective rule mining using genetic algorithms (MOEA_Ghosh) [19]: This algorithm uses a Pareto based GA to extract some useful and interesting rules from any dataset. Each chromosome represents an association rule, where for each attribute bits are associated that indicate the antecedent or consequent of the rule, the absence or presence of the attribute, and the relational operators involved with the attribute. It uses three measures: comprehensibility, interestingness, and predictive accuracy to solve the multiobjective rule mining problem. A separate population is used, which will contain those chromosomes that are non-dominated from among the current population, as well as the non-dominated solutions from the previous generation.
8) Apriori [6], [27]: Apriori follows a breadth-first strategy. It generates candidate itemsets for the current iteration by means of itemsets considered to be frequent in the previous iteration. It then enumerates all the subsets for each transaction and increments the support of candidates matching them. Then, those that have the user-specified minSup, are marked as frequent for the next iteration. This process is repeated until all frequent itemsets have been found. Finally, Apriori uses the frequent itemsets to generate positive rules with confidence greater than minConf.
9) Eclat [28]: Eclat employs a depth-first strategy. It generates candidates by extending the prefixes of an itemset until an infrequent one is found. In such cases, it simply backtracks to the previous prefix and then recursively applies the above procedure. Unlike Apriori, for all the items in a dataset, it first constructs a list of all the transaction identifiers (tid-list) containing that item. Then it counts the support by merely intersecting two or more tid-lists to check whether they have items in common. If so, the support is equal to the size of the resulting set. The process for generating the positive rules is the same as Apriori.
10) Nondominated Sorting Genetic Algorithm II (NSGA-II) [29]: This MOEA is one of the most well known and frequently used in the literature. NSGA-II uses an evolutionary model similar to other evolutionary algorithms but with two different features, which make it a high-performance MOEA. One is the fitness evaluation of each solution based on Pareto ranking and a crowding measure, and the other is an elitist generation update procedure.

TABLE II
PARAMETERS CONSIDERED FOR THE COMPARISON

Algorithms	*Parameters*
Alatasetal	N_{eval}=50000, nInitialRandomChromo=12, r = 3, TournamentSize = 10, P_{sel}= 0.25, P_{cro} = 0.7, P_{mut_min} = 0.05, P_{mut_max} = 0.9, W_{sup} = 5, W_{conf} = 20, $W_{amplRule}$ = 0.05, $W_{amplInterv}$ = 0.02, $W_{covered}$ = 0.01
EARMGA	PopSize = 100, N_{eval} = 50000, k = 2 (3 with HH), P_{sel} = 0.75, P_{cro} = 0.7, P_{mut} = 0.1, α = 0.01
GAR	PopSize = 100, nItemset = 100, N_{eval} = 50000, P_{sel}= 0.25, P_{cro} = 0.7, P_{mut} = 0.1, ω = 0.4, Ψ= 0.7, μ= 0.5, minSup = 0.1, minConf = 0.8
GENAR	PopSize = 100, N_{eval} = 50000, P_{sel}= 0.25, P_{cro} = 0.7, P_{mut} = 0.1, nRules = 30, FP = 0.7, AF = 0.2
ARMMGA	PopSize=100, N_{eval}=50000, k=2 (3 with HH), P_{sel}=0.95, P_{cro}=0.85, P_{mut} = 0.01, db=0.01
MODENAR	PopSize = 100, N_{eval}=50000, Threshold= 60, CR = 0.3, W_{sup} = 0.8, W_{conf} = 0.2, W_{comp} = 0.1, $W_{amplInterv}$ = 0.4
MOEA_Ghosh	PopSize = 100, N_{eval}=50000, PointCrossover=2, P_{cro}=0.8, P_{mut}= 0.02
Apriori	minSup = 0.1, minConf = 0.8
Eclat	minSup = 0.1, minConf = 0.8
NSGAII	PopSize = 100, N_{eval}=50000, γ=2, P_{mut}= 0.1
MOPNAR	N_{eval}=50000, H=13, m=3, PopSize=N_{H+m-1}^{m-1}, T=10, δ=0.9, η_r=2, γ=2, P_{mut}= 0.1, α = 5%

TABLE III
RESULTS FOR ALL DATASETS IN THE COMPARISON WITH ALATASETAL

$Algorithm$	$\#R$	Av_{Sup}	Av_{Conf}	Av_{Lift}	Av_{Conv}	Av_{CF}	$Av_{Netconf}$	$Av_{Yule'sQ}$	Av_{Amp}	$\%Tran$
Bolts										
Alatasetal	29.6	**0.95**	**1**	1.03	∞	0.12	0.12	0.12	3.65	95
MOPNAR	**53.6**	0.34	**1**	**14.82**	∞	**0.99**	**0.95**	**1**	**2.29**	**100**
Flare										
Alatasetal	**86.8**	0.13	**1**	1.38	∞	**0.91**	0.15	0.64	8.16	92.31
MOPNAR	31.8	**0.4**	0.91	**8.24**	∞	0.88	**0.58**	**0.93**	**2.94**	**100**
House16H										
Alatasetal	90.67	0.19	**0.99**	1.03	∞	0.58	0.03	0.41	8.76	98.09
MOPNAR	**99**	**0.31**	0.95	**10.23**	∞	**0.92**	**0.78**	**0.99**	**2.7**	**99.96**
Movement Libras										
Alatasetal	0	-	-	-	-	-	-	-	-	-
MOPNAR	**53.6**	**0.24**	**0.97**	**16.62**	∞	**0.96**	**0.92**	**1**	**2.49**	**94.28**
Pollution										
Alatasetal	34.6	**0.59**	**1**	6.3	∞	0.43	0.39	0.43	3.18	59.67
MOPNAR	**45**	0.26	0.98	**23.58**	∞	**0.96**	**0.81**	**0.99**	**2.45**	**99**
Quake										
Alatasetal	4.25	**0.67**	**1**	1.01	∞	0.1	0	0	**2.08**	98.06
MOPNAR	**54.6**	0.27	0.91	**6.42**	∞	**0.84**	**0.55**	**0.94**	2.32	**99.2**
Segment										
Alatasetal	47.5	**0.53**	0.94	1.03	∞	0.26	0.02	0.21	4.14	**100**
MOPNAR	**86.8**	0.3	**0.98**	**14.4**	∞	**0.98**	**0.92**	**1**	**2.57**	**100**
Stock										
Alatasetal	14.4	0.08	**1**	**96.54**	∞	0.92	0.72	0.75	**2.73**	21.04
MOPNAR	**82.4**	**0.31**	0.94	8.17	∞	**0.93**	**0.81**	**1**	2.75	**99.6**
Stulong										
Alatasetal	10	**0.61**	**0.99**	2.59	∞	0.39	0.21	0.32	2.97	99.25
MOPNAR	**73.6**	0.27	0.82	**4.29**	∞	**0.74**	**0.52**	**0.93**	**2.83**	**99.85**

The parameters of the analyzed algorithms are shown in Table II. With these values, for our proposal, we have tried to facilitate comparisons, selecting standard common parameters that work well in most cases instead of searching for very specific values. The parameters of the remaining algorithms were selected according to the recommendations of the corresponding authors of each proposal, which are the default parameter settings included in the KEEL software tool [43].

Notice that the length of the rules for EARMGA and ARMMGA is higher in the dataset House_16H, because the number of attributes and transactions is higher in this problem. Moreover, only Apriori, Eclat and GAR need a minSup and a minConf to mine positive QARs. In order to facilitate comparisons, we have selected 0.1 for minSup and 0.8 for minConf for all the datasets, which are standard common values that work well in most cases, instead of searching for highly specific values for each one. For all the experiments conducted in this paper, the results shown in the tables for the multiobjective algorithms always refer to nondominated association rules.

C. Comparison With PNQARs Alatas's Algorithm

In this section we study the performance of our proposal against an evolutionary approach for mining PNQARs, the Alatasetal algorithm [9]. The results obtained by the analyzed algorithms are shown in Table III, where #*R* is the number

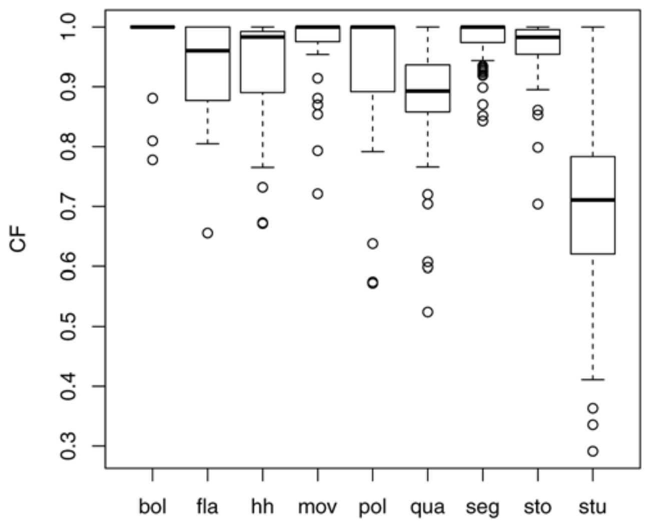


Fig. 4. Boxplot of the CF measure for all datasets.

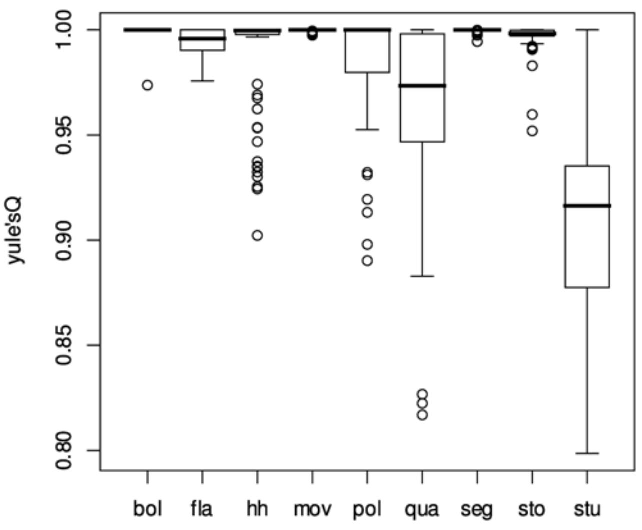


Fig. 5. Boxplot of the yule'sQ measure for all datasets.

of generated association rules, Av_{Sup} and Av_{Conf} are, respectively, the average support and the average confidence of the rules, Av_{Lift} is the average value for the lift measure of the rules, Av_{Conv} is the average value for the conviction measure of the rules, Av_{CF} is the average value for the CF measure of the rules, $Av_{Netconf}$ is the average value for the netconf measure of the rules, $Av_{Yule'sQ}$ is the average value for the yule'sQ measure of the rules, Av_{Amp} is the average length of the rules in terms of attributes involved, and $\%Tran$ is the percentage of examples covered by the rules. The value ∞ shown in the table represents the maximum value for some measures (see Section II).

We can present the following conclusions from an analysis of the results presented in Table III.

1) The rules obtained by our proposal present improvements for almost all the interestingness measures over those obtained by Alatasetal in all the datasets. Alatasetal extracts rules with good average support and confidence but some of them present high-support itemsets in the consequent or negative dependencies, obtaining low values for the rest of the measures.
2) Our proposal obtains a reduced set of short PNQARs without overlapping rules (less than 100 in all the datasets) where each rule provides us with interesting knowledge of the dataset. Moreover, the coverage of the dataset is very high (close to 100% in all the datasets), providing us with knowledge of the whole dataset. Alatasetal obtains a smaller number of rules than our proposal but with lower values of coverage for allmost all the datasets.
3) MOPNAR presents a reduced set of interesting PNQARs, obtaining a good trade-off between the number of rules, support and coverage.

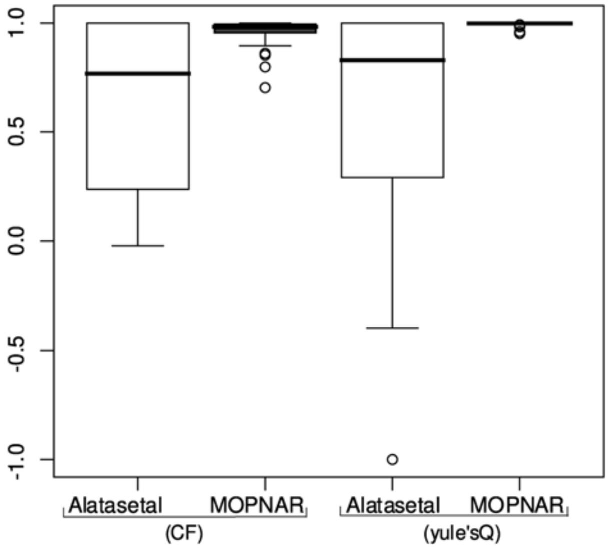


Fig. 6. Boxplot of the CF and yule'sQ measures for Alatas's algorithm and our proposal in the dataset Stock.

Figs. 4 and 5 are boxplot graphics that show values for the measures CF and yule'sQ, respectively, for the rules obtained from one of the five runs performed by our proposal for all the datasets, selected at random. We can see how all the rules represent positive dependencies with values close to maximum values for these measures (see Section II). Notice that more than 75% of the rules obtained have a value greater than 0.85 for the CF (less in Stulong) and for the yule'sQ. Fig. 6 shows the values obtained by Alatasetal and our proposal for the measures CF and yule'sQ in the dataset Stock. We can see that MOPNAR presents better values of CF and yule'sQ than Alatasetal, as all the rules obtain values close to the maximum values for these measures. Notice how some rules obtained by Alatasetal represent independence or negative dependence according to these measures.

D. Comparison With Other Evolutionary Algorithms

This section compares the performance of our algorithm with three mono-objective algorithms (EARMGA [15], GAR [26], and GENAR [14]) and three MOEAs for mining QARs (ARMMGA [23], MODENAR [18], MOEA_Ghosh [19]).

TABLE IV

RESULTS OBTAINED BY EVOLUTIONARY ALGORITHMS

$Algorithm$	$\#R$	Av_{Sup}	Av_{Conf}	Av_{Lift}	Av_{Conv}	Av_{CF}	$Av_{Netconf}$	$Av_{Yule'sQ}$	Av_{Amp}	$\%Tran$
Bolts										
EARMGA	**100**	0.35	**1**	1.10	∞	0.20	0.07	0.20	**2**	**100**
GAR	33.20	0.21	0.98	4.21	∞	0.96	0.86	0.99	3.36	91.50
GENAR	30	0.14	**1**	1.57	∞	**1**	0.42	**1**	8	39
ARMMGA	1	0.46	**1**	1.3	∞	**1**	0.42	**1**	**2**	46
MODENAR	39.60	0.39	0.94	2.27	∞	0.48	0.41	0.54	3.93	68.5
MOEA_Ghosh	11.8	**0.76**	0.95	4.08	∞	0.33	0.29	0.37	6.33	**100**
MOPNAR	53.6	0.34	**1**	**14.82**	∞	0.99	**0.95**	**1**	2.29	**100**
Flare										
EARMGA	95.8	0.49	**1**	1.01	∞	0.01	0.01	0	**2**	**100**
GAR	**380.6**	**0.93**	0.99	1.01	∞	0.44	0.15	0.86	3.86	99.65
GENAR	30	0.06	0.97	6.26	∞	**0.97**	**0.85**	**0.99**	12	32.84
ARMMGA	1	0.88	**1**	1.01	4.13	0.66	0.12	0.82	**2**	86.95
MODENAR	28.6	0.29	0.86	1.91	∞	0.49	0.2	0.44	6.62	98.94
MOEA_Ghosh	19	0.59	0.84	1.71	∞	0.52	0.31	0.66	6.34	92.37
MOPNAR	31.8	0.4	0.91	**8.24**	∞	0.88	0.58	0.93	2.94	**100**
House16H										
EARMGA	60.2	0.17	**1**	1.01	∞	0.28	0.01	0.01	3	98.56
GAR	**105.6**	0.76	0.9	1.03	1.33	0.2	0.17	0.48	**2.01**	**99.99**
GENAR	30	0.44	0.99	1.02	2.09	0.44	0.03	0.41	17	87.31
ARMMGA	1	**0.98**	0.99	1.01	1.13	0.09	0.04	0.16	3	97.97
MODENAR	64.40	0.69	0.99	1.15	∞	0.72	0.19	0.74	7	81
MOEA_Ghosh	19.8	0.6	0.79	**269.17**	∞	0.36	0.11	0.3	7.75	98.87
MOPNAR	99	0.31	0.95	10.23	∞	**0.92**	**0.78**	**0.99**	2.7	99.96
Movement Libras										
EARMGA	**100**	0.39	**1**	1	∞	0	0	0	**2**	**100**
GAR	2.60	**0.42**	0.94	3.73	12.62	0.90	0.90	**1**	**2**	53.28
GENAR	30	0.04	0.89	13.35	∞	0.89	0.86	0.99	91	53.73
ARMMGA	1	0.26	0.87	3.27	28.86	0.79	0.79	0.93	**2**	25.95
MODENAR	23.4	0.01	0.15	32.31	∞	0.04	0.15	0.17	61.61	3.17
MOEA_Ghosh	10.8	0.01	0.22	**79.47**	∞	0.22	0.22	0.22	80.08	0.28
MOPNAR	53.6	0.24	0.97	16.62	∞	**0.96**	**0.92**	**1**	2.49	94.28
Pollution										
EARMGA	**100**	0.25	**1**	1.19	∞	0.31	0.06	0.27	**2**	**100**
GAR	54	**0.67**	0.91	1.17	∞	0.53	0.43	0.77	2.03	**100**
GENAR	30	0.22	**1**	1.23	∞	0.98	0.24	0.98	16.00	48
ARMMGA	1	0.64	**1**	1.04	∞	**1**	0.10	**1**	**2**	63.34
MODENAR	34.40	0.27	0.91	2.94	∞	0.85	0.52	0.94	7.20	48.34
MOEA_Ghosh	26.8	0.18	0.67	8.89	∞	0.61	0.62	0.95	13.18	39
MOPNAR	45	0.26	0.98	**23.58**	∞	0.96	**0.81**	0.99	2.45	99
Quake										
EARMGA	**100**	0.27	**1**	1	∞	0.01	0	0	**2**	**100**
GAR	1	0.44	0.84	0.98	0.88	-0.03	-0.05	-0.17	**2**	52.89
GENAR	30	0.55	0.95	1.01	1.09	0.09	0.02	0.10	4	81.78
ARMMGA	1	0.66	0.73	1.01	1.02	0.02	0.04	0.09	**2**	65.94
MODENAR	63.60	0.36	0.84	**117.09**	∞	0.31	0.09	0.22	2.09	92.84
MOEA_Ghosh	8	**0.86**	**1**	1.01	∞	0.18	0.01	0.14	3.09	**100**
MOPNAR	54.6	0.27	0.91	6.42	∞	**0.84**	**0.55**	**0.94**	2.32	99.2
Segment										
EARMGA	**99.20**	0.45	**1**	1.04	∞	0.08	0.02	0.04	**2**	**100**
GAR	18.8	0.36	0.89	2.49	3.61	0.58	0.47	0.73	**2**	97.97
GENAR	30	0.07	0.78	5.43	∞	0.74	0.70	0.93	20	83.49
ARMMGA	1	**0.92**	**1**	1.14	∞	0.2	0.2	0.2	**2**	91.33
MODENAR	58.80	0.33	0.97	1.72	∞	0.93	0.58	0.96	10.60	56.49
MOEA_Ghosh	28.2	0.36	0.86	**108.6**	∞	0.73	0.5	0.8	12.63	72.95
MOPNAR	86.8	0.3	0.98	14.4	∞	**0.98**	**0.92**	**1**	2.57	**100**
Stock										
EARMGA	**100**	0.37	**1**	1.01	∞	0.02	0.01	0.02	**2**	**100**
GAR	2	0.56	0.87	1.35	3.19	0.62	0.62	0.88	**2**	73.30
GENAR	30	0.29	0.92	1.69	∞	0.81	0.54	0.89	10	88.51
ARMMGA	1	0.37	0.77	1.63	2.25	0.56	0.56	0.86	**2**	36.22
MODENAR	63.80	0.48	0.92	1.75	∞	0.61	0.30	0.54	3	81.86
MOEA_Ghosh	19.8	**0.61**	0.91	**42.56**	∞	0.53	0.36	0.68	5.28	96.4
MOPNAR	82.4	0.31	0.94	8.17	∞	**0.93**	**0.81**	**1**	2.75	99.6
Stulong										
EARMGA	92.6	0.27	**1**	1.01	∞	0.13	0.01	0.02	**2**	**100**
GAR	**157.4**	0.78	0.94	1.03	1.63	0.31	0.21	0.63	2.96	99.94
GENAR	30	**0.88**	0.99	1.01	1.02	0.02	0.01	0.04	5	95.26
ARMMGA	1	0.87	0.87	1.01	1.03	0.03	**0.62**	0.91	**2**	86.38
MODENAR	63.20	0.52	0.88	**13.94**	∞	0.27	0.06	0.27	3	99.28
MOEA_Ghosh	19.6	0.83	0.99	1.04	∞	0.51	0.23	0.6	3.47	99.92
MOPNAR	73.6	0.27	0.82	4.29	∞	**0.74**	0.52	**0.93**	2.83	99.85

TABLE V

RESULTS FOR APRIORI AND ECLAT WITH THREE, FOUR, AND FIVE INTERVALS PER ATTRIBUTE

$Algorithm$	$\#R$	Av_{Sup}	Av_{Conf}	Av_{Lift}	Av_{Conv}	Av_{CF}	$Av_{Netconf}$	$Av_{Yule'sQ}$	Av_{Amp}	$\%Tran$
					House16H					
Apriori-3	2382102	0.25	0.98	1.93	∞	0.81	0.38	0.76	9.05	100
Apriori-4	1749917	0.22	0.97	2.19	∞	0.83	0.45	0.76	8.65	100
Apriori-5	1073035	0.2	0.96	2.9	∞	0.85	0.56	0.8	8.26	100
Eclat-3	2382102	0.25	0.98	1.93	∞	0.81	0.38	0.76	9.05	100
Eclat-4	1749917	0.22	0.97	2.19	∞	0.83	0.45	0.76	8.65	100
Eclat-5	1073035	0.2	0.96	2.9	∞	0.85	0.56	0.8	8.26	100
					Stulong					
Apriori-3	99	0.35	0.92	2.02	∞	0.58	0.41	0.62	3.33	100
Apriori-4	89	0.31	0.93	1.22	∞	0.43	0.14	0.29	3.26	99.86
Apriori-5	44	0.25	1	1.01	∞	0.33	0.01	0.24	3.12	98.81
Eclat-3	99	0.35	0.92	2.02	∞	0.58	0.41	0.62	3.33	100
Eclat-4	89	0.31	0.93	1.22	∞	0.43	0.14	0.29	3.26	99.86
Eclat-5	44	0.25	1	1.01	∞	0.33	0.01	0.24	3.12	98.81

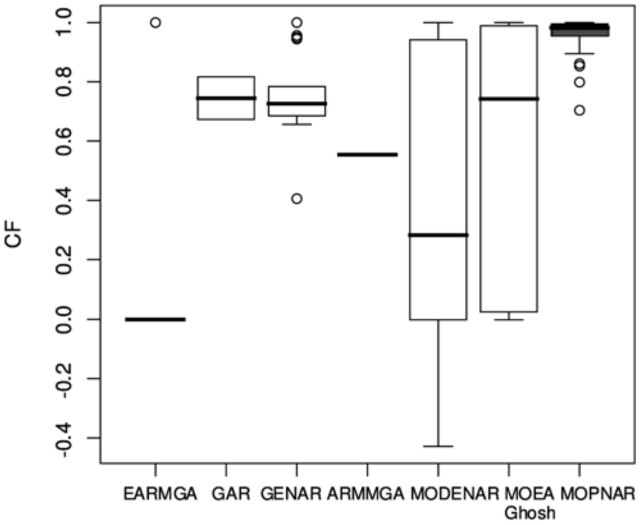


Fig. 7. Boxplot of the CF measure for the evolutionary algorithms in the dataset Stock.

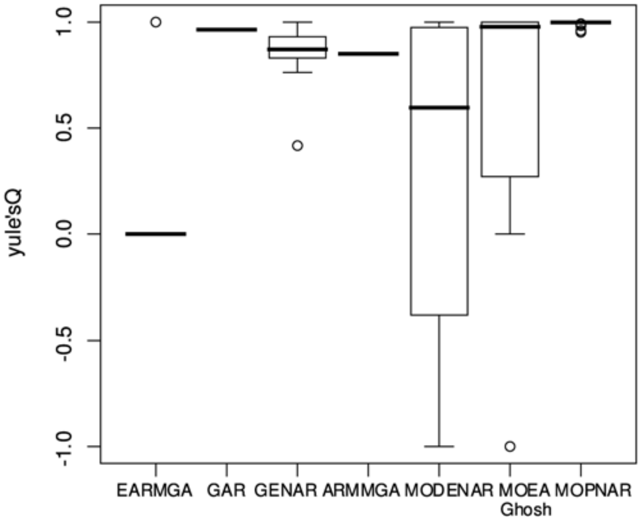


Fig. 8. Boxplot of the yule'sQ measure for the evolutionary algorithms in the dataset Stock.

The results obtained by the analyzed algorithms are shown in Table IV (this kind of table was described in Section IV-C). Through the analysis of the results presented in these tables, we can highlight the following facts.

1) The rules obtained by our proposal present values for the interestingness measures that are better than or similar to the rules obtained by the analyzed algorithms in all the datasets. As with Alataetal, some of these algorithms obtain good average support and confidence but the average values for the rest of the measures are low in some datasets due to the fact that they obtain rules with high-support itemsets in the consequent or negative dependences.
2) MOPNAR allows us to mine reduced sets of interesting PNQARs, which provide us with interesting knowledge of the whole datasets, presenting average values of coverage greater than 99% in all cases, while the rest of the analyzed algorithms obtain values worse than or similar to MOPNAR. Notice that EARMGA obtains the best coverage for almost all of the datasets but its rules present low values for the interestingness measures, and that the GENAR obtains low values of coverage for almost all of the datasets due to the fact that the rules obtained always involve all the attributes in the dataset.
3) Finally, the rules obtained by MOPNAR have a low number of attributes for all the datasets, giving the advantage of easier understanding from a user's perspective.

Figs. 7 and 8 are boxplots that show the values for the measures CF and yule'sQ for the rules obtained from one of the five runs performed by all the evolutionary algorithms analyzed in the dataset stock, selected at random. We can see that MOPNAR presents the best values of CF and yule'sQ in comparison with the rest of the algorithms, with almost all of the rules obtaining values greater than 0.9 for these measures. Notice how some rules obtained by MODENAR and EARMGA represent independence or negative dependece according to these measures.

E. Comparison With Classical Algorithms

This section compares the performance of our algorithm with two classical association rules extraction algorithms,

TABLE VI

RESULTS FOR ALL DATASETS IN THE COMPARISON WITH CLASSICAL ALGORITHMS FOR MINING ASSOCIATION RULES

$Algorithm$	$\#R$	Av_{Sup}	Av_{Conf}	Av_{Lift}	Av_{Conv}	Av_{CF}	$Av_{Netconf}$	$Av_{Yule'sQ}$	Av_{Amp}	$\%Tran$
Bolts										
Apriori	**1246**	0.15	0.99	7.16	∞	0.98	**0.96**	**1**	4.36	97.50
Eclat	**1246**	0.15	0.99	7.16	∞	0.98	**0.96**	**1**	4.36	97.50
NSGA-II	80.2	0.33	**1**	7.75	∞	**0.99**	0.93	**1**	2.55	**100**
MOPNAR	53.6	**0.34**	**1**	**14.82**	∞	**0.99**	0.95	**1**	**2.29**	**100**
Flare										
Apriori	**28512**	0.18	**0.97**	4.95	∞	**0.94**	**0.82**	0.97	5.87	**100**
Eclat	**28512**	0.18	**0.97**	4.95	∞	**0.94**	**0.82**	0.97	5.87	**100**
NSGA-II	63.6	0.33	0.9	**45.38**	∞	0.83	0.51	0.87	3.59	98.65
MOPNAR	31.8	**0.4**	0.91	8.24	∞	0.88	0.58	**0.93**	**2.94**	**100**
House16H										
Apriori	**1749917**	0.22	**0.97**	2.19	∞	0.83	0.45	0.76	8.65	**100**
Eclat	**1749917**	0.22	**0.97**	2.19	∞	0.83	0.45	0.76	8.65	**100**
NSGA-II	99.6	**0.37**	0.95	**419.07**	∞	0.91	0.57	0.88	3.73	98.32
MOPNAR	99	0.31	0.95	10.23	∞	**0.92**	**0.78**	**0.99**	**2.7**	99.96
Movement Libras										
Apriori	-	-	-	-	-	-	-	-	-	-
Eclat	-	-	-	-	-	-	-	-	-	-
NSGA-II	**63.40**	**0.24**	0.96	**29.49**	∞	0.94	0.91	**1**	**2.38**	90.17
MOPNAR	53.60	**0.24**	**0.97**	16.62	∞	**0.96**	**0.92**	**1**	2.49	**94.28**
Pollution										
Apriori	**41510**	0.13	0.95	5.84	∞	0.93	**0.86**	0.98	5.88	**100**
Eclat	**41510**	0.13	0.95	5.84	∞	0.93	**0.86**	0.98	5.88	**100**
NSGA-II	26.8	0.2	0.97	**30.21**	∞	0.95	0.85	**0.99**	**2**	95.67
MOPNAR	45	**0.26**	**0.98**	23.58	∞	**0.96**	0.81	**0.99**	2.45	99
Quake										
Apriori	18	0.25	0.91	1	1.15	0.11	-0.01	0.02	2.56	90.55
Eclat	18	0.25	0.91	1	1.15	0.11	-0.01	0.02	2.56	90.55
NSGA-II	**96.8**	0.24	**0.92**	**59.67**	∞	**0.88**	**0.64**	**0.94**	2.82	92
MOPNAR	54.6	**0.27**	0.91	6.42	∞	0.84	0.55	**0.94**	**2.32**	**99.2**
Segment										
Apriori	-	-	-	-	-	-	-	-	-	-
Eclat	-	-	-	-	-	-	-	-	-	-
NSGA-II	**86.80**	0.29	**0.98**	**73.30**	∞	0.97	0.84	0.97	2.62	98.16
MOPNAR	**86.80**	**0.30**	**0.98**	14.40	∞	**0.98**	**0.92**	**1**	**2.57**	**100**
Stock										
Apriori	**855**	0.13	0.91	4.77	∞	0.88	0.76	0.96	4.16	99.48
Eclat	**855**	0.13	0.91	4.77	∞	0.88	0.76	0.96	4.16	99.48
NSGA-II	100	0.22	**0.94**	**19.97**	∞	**0.93**	**0.87**	**1**	3.12	96.97
MOPNAR	82.4	**0.31**	**0.94**	8.17	∞	**0.93**	0.81	**1**	**2.75**	**99.6**
Stulong										
Apriori	89	**0.31**	**0.93**	1.22	∞	0.43	0.14	0.29	3.26	**99.86**
Eclat	89	**0.31**	**0.93**	1.22	∞	0.43	0.14	0.29	3.26	**99.86**
NSGA-II	**95.8**	**0.31**	0.89	**13.33**	∞	**0.81**	**0.53**	**0.93**	3.29	98.1
MOPNAR	73.6	0.27	0.82	4.29	∞	0.74	0.52	**0.93**	**2.83**	99.85

Apriori [6], [27], and Eclat [28], and with the classical MOEA NSGA-II [29]. A commonly used method to mine QARs from classical association rules extraction algorithms is to partition the domains, introducing new attributes with intervals. In this paper, for each quantitative attribute we have used a uniform partition [44], which is the usual discretization algorithm applied when we lack additional information with which to apply algorithms based on information theory [45], [46] or other concepts [47].

The problem is finding an appropriate number of intervals for a quantitative attribute. This problem was first introduced in [6], in which the authors pointed out that if too many intervals are defined for the attributes, the rules obtained might not achieve the minimum support threshold. On the other hand, if intervals are defined that are too large, the rules might not achieve the confidence threshold. For this reason, several experiments have been carried out with three, four and five intervals per attribute, in order to select the number of intervals. Table V shows the results obtained by Apriori and Eclat (this kind of table was described in Section IV-C). Analyzing the results obtained in House_16H, we can see that $\#R$, Av_{Sup}, and Av_{Conf} decrease in accordance with the increase in the number of intervals. Moreover, in Stulong, the number of rules that can achieve the confidence threshold decreases quickly with the increase in the number of intervals, while the values for the interesting measures present the same behavior. Therefore, we will use a uniform partition with four intervals for each quantitative attribute in the experiments.

In order to compare our proposal with NSGA-II, we have used the same code scheme, objectives, initial gene pool and

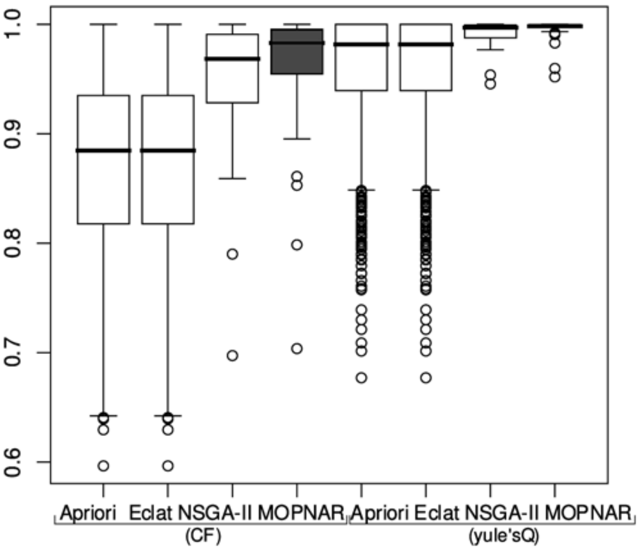


Fig. 9. Boxplot of the CF and yule'sQ measures for classical algorithms in the dataset stock.

genetic operators as in our proposal. The results obtained by the classical algorithms are shown in Table VI (this kind of table was described in Section IV-C). Analyzing the results presented in this table we can see how.

1) In most datasets Apriori and Eclat mine large sets of QARs, obtaining a good coverage of the datasets. Notice that movement libras and segment have a large number of attributes and they cannot be run. By contrast, our proposal allows us to obtain a reduced set of PNQARs with a good coverage in all datasets. Moreover, the rules obtained present values for the interestingness measures that are better than or similar to the rules obtained by Apriori and Eclat in all the datasets. Notice that, in the dataset Quake, Apriori and Eclat extract rules with good average support and confidence but low values for the rest of the measures due to the fact that they first need to partition the quantitative domains into intervals in order to mine the QARs, and these partitions are not appropriate for this dataset.
2) NSGA-II obtains reduced sets of PNQARs with good values for the interestingness measures; however, in two datasets the number of rules obtained by NSGA-II is limited by the size of the population (100 rules) and the coverage is less than that of our proposal in all datasets. Moreover, MOPNAR obtains PNQAR sets with a smaller number of rules in almost all of the datasets, and with similar values for the interestingness measures.

Fig. 9 presents a boxplot that shows the values for the measures CF and yule'sQ for the rules obtained from the classical algorithms and MOPNAR in the dataset stock. We can see that all the rules obtained represent positive dependencies, and that MOPNAR and NSGA-II obtain rules with values close to each other. Moreover, MOPNAR presents better values of CF and yule'sQ than the classical algorithms, but it and NSGA-II obtain rules with values close to the maximum values for these measures.

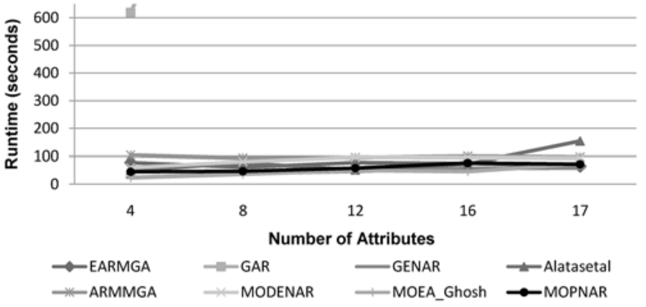


Fig. 10. Relationship between the runtime and the number of attributes with the datasets House_16H for the evolutionary algorithms.

TABLE VII
RUNTIME (SECONDS) EXPENDED BY ALL THE ALGORITHMS WHEN THE NUMBER OF ATTRIBUTES INCREASES WITH THE DATASETS HOUSE_16H

	Number of Attributes				
Algorithms	4	8	12	16	17
EARMGA	78	61	78	75	65
GAR	619	1528	1991	2083	2014
GENAR	24	38	**47**	57	**59**
Alatasetal	50	77	52	72	154
ARMMGA	104	93	95	100	97
MODENAR	60	81	97	95	93
MOEA_Ghosh	23	36	50	**46**	73
Apriori	**3**	**5**	233	5192	11268
Eclat	**3**	**5**	251	5812	12467
NSGA-II	54	56	55	96	77
MOPNAR	45	47	58	77	72

TABLE VIII
RUNTIME (SECONDS) EXPENDED BY ALL THE ALGORITHMS WHEN THE NUMBER OF EXAMPLES INCREASES WITH THE DATASETS HOUSE_16H

	Number of Examples				
Algorithms	20%	40%	60%	80%	100%
EARMGA	15	31	39	50	65
GAR	467	898	1260	1595	2014
GENAR	**12**	**23**	**36**	47	**59**
Alatasetal	16	24	93	104	154
ARMMGA	21	40	60	77	97
MODENAR	13	41	37	115	93
MOEA_Ghosh	13	24	37	**44**	73
Apriori	2689	5180	10050	9004	11268
Eclat	3076	8700	11300	10604	12467
NSGA-II	13	27	51	67	77
MOPNAR	16	31	47	59	72

F. Analysis of Complexity and Scalability

Several experiments have been carried out to analyze the scalability of the algorithms in the dataset House_16H. All of the experiments were performed using an Intel Core i7, 2.80 GHz CPU with 12 GB of memory and running Linux. The average runtime expended by the analyzed algorithms when the number of attributes and examples increases are shown in Tables VII and VIII, respectively.

Figs. 10 and 11 show the relationship between the runtime and the number of attributes for all algorithms studied. We can see how almost all the evolutionary algorithms scale quite linearly when the number of attributes in the dataset increases, however the classical association rules extraction algorithms

TABLE IX

RULES OBTAINED BY OUR PROPOSAL FROM SEVERAL DATASETS

Data	*Rule*	*Sup*	*Conf*	*CF*	*YulesQ*
Bolts	R1: If **SENS** is not $[7.0, 10.0]$ then **SPEED1** is not $[3.0, 5.0]$	0.49	1	1	1
Flare	R2: If **HistComplex** is not 2 then **X-class** is 0	0.59	1	1	1
Quake	R3: If **Latitude** is $[-10.58, 47.44]$ then **Longitude** is not $[-179.96, -155.45]$	0.59	1	1	1
Stock	R4: If **Company2** is not $[46.38, 56.99]$ then **Company1** is not $[17.22, 31.99]$	0.58	0.99	0.97	0.99

TABLE X

RELATIONSHIP BETWEEN PNQARS OBTAINED BY OUR PROPOSAL AND POSITIVE QARS OBTAINED BY OTHER ALGORITHMS

Data	*Our proposal*	*Other algorithms*
Bolts	If **SENS** is $[0, 6]$ then **SPEED2** is not $[1.51, 2.49]$	If **SENS** is $[0, 6]$ then **SPEED2** is $[0, 1.5]$ If **SENS** is $[0, 6]$ then **SPEED2** is $[2.5, 2.5]$
Quake	If **Focal depth** is not $[136, 176]$ and If **Longitude** is $[-179.88, -171.16]$ then **Latitude** is $[-33.79, -14.91]$	If **Focal depth** is $[0, 135]$ and **Longitude** is $[-179.88, -171.16]$ then **Latitude** is $[-33.79, -14.91]$ If **Focal depth** is $[177, 656]$ and **Longitude** is $[-179.88, -171.16]$ then **Latitude** is $[-33.79, -14.91]$
Stock	If **Company2** is not $[46.38, 56.99]$ then **Company1** is not $[17.22, 31.99]$	If **Company2** is $[19.25, 46.37]$ then **Company1** is $[32, 61.5]$ If **Company2** is $[57, 60.25]$ then **Company1** is $[32, 61.5]$

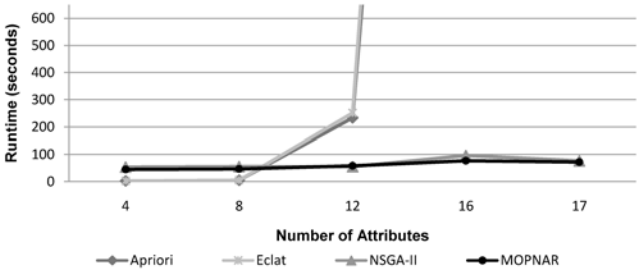


Fig. 11. Relationship between the runtime and the number of attributes with the datasets House_16H for the classical algorithms.

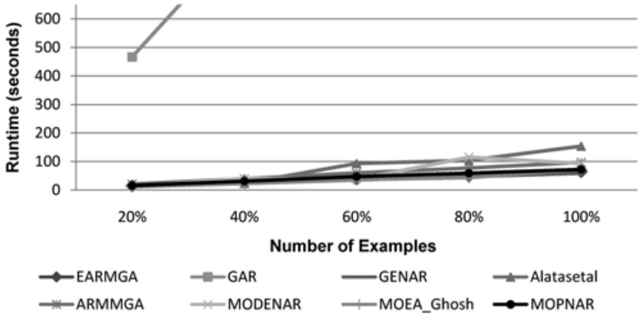


Fig. 12. Relationship between the runtime and the number of examples with the datasets House_16H for the evolutionary algorithms.

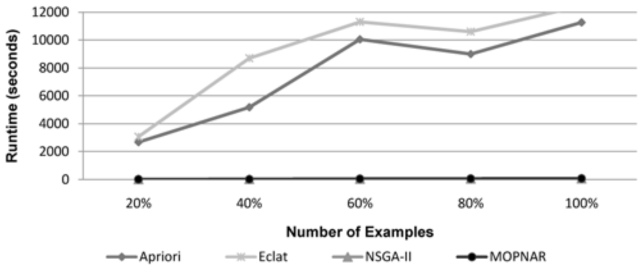


Fig. 13. Relationship between the runtime and the number of examples with the datasets House_16H for the classical algorithms.

(Apriori and Eclat) increases exponentially when the number of attributes is higher than ten. Moreover, we can see that the GAR expends a large amount of time mining the association rules because it needs an additional process to extract the association rules.

Figs. 12 and 13 show the relationship between the runtime and the number of examples. As in the previous case, the runtime scales quite linearly when the number of examples in the dataset increases. Moreover, we can see how the increase in the number of examples and attributes affects classical association rules extraction algorithms more than the evolutionary algorithms. Notice that Figs. 10 and 12 show few results pertaining to the GAR because its runtime exceeds more than 650 s in almost all cases, which can easily be seen from Tables VII and VIII.

G. Some Rules Obtained by Our Proposal

In this section, we analyze some PNQARs mined by the proposed approach. Table IX shows some interesting PNQARs obtained from several datasets, where *Data* is the dataset in which the rule was obtained, *Rule* is the rule obtained, *Sup* and *Conf* are, respectively, the support and the confidence of the rules, *CF* is the value for the CF measure of the rules, and *YulesQ* is the value for the yule'sQ measure of the rules.

These rules could be interpreted as follows. The Bolts dataset was generated to store data from experiments on the effects of machine adjustments on the time taken to count bolts. $R1$ indicates that when the sensitivity of the electronic eye (SENS) is less than seven (the domain of this variable is [0, 10]) then the speed of rotation (SPEED1) of the plate is lower than three (low) or higher than five (very fast). The Flare dataset stores data on the number of times a certain type of solar flare occured in a 24 hour period. $R2$ shows that when the region is not historically complex (HistComplex) then severe flares (X-class) will not be produced in the following 24 hours. The Quake dataset provides data on smoothing methods in statistics. In this case, $R3$ indicates that when the latitude is between -10.58 and 47.44 then longitude is lower than

-155.45. The data provided by the Stock dataset are daily stock prices from January 1988 through October 1991, for ten aerospace companies. *R*4 shows that when the second company (Company2) does not have prices between 46.38 and 56.99 then the first company (Company1) does not have prices between 17.22 and 31.99.

Most of these rules obtained by our proposal include negated items, allowing us to reduce the number of rules needed to extract interesting knowledge from datasets. In order to illustrate this fact, Table X shows some PNQARs obtained by our proposal and the positive QARs obtained by other analyzed algorithms that were needed to extract the same knowledge, where *Data* is the dataset in which the rules were obtained, *Our proposal* is the PNQARs obtained by our proposal and *Other algorithms* are the positive QARs obtained by other algorithms.

V. Conclusion

We have proposed MOPNAR, a new MOEA that allows mining with a reduced set of PNQARs. The PNQARs are easy to understand, interesting, and offer good coverage of the dataset, maximizing three objectives: comprehensibility, interestingness, and performance. To accomplish this, this proposal extends the MOEA MOEA/D-DE to perform an evolutionary learning of the intervals of attributes and a condition selection for each rule. This proposal introduces an EP and a restarting process to the evolutionary model in order to store all the nondominated rules found and to improve the diversity of the rule set obtained. Moreover, the rules obtained are very strong, which indicates a strong dependence between the items and solves the support drawback.

Taking into account the results obtained over nine real-world datasets, we can conclude that our proposal allows us to mine PNQAR sets with a good trade-off between the number of rules, support, and coverage, presenting high coverages in all the datasets. Moreover, MOPNAR obtains reduced sets of PNQARs with few attributes, making it easier to understand from a user's perspective, and with high values for the interestingness measures in all datasets. Finally, the proposed approach presents a good computational cost in all datasets and good scalability when the size of the problem increases.

References

[1] R. Renesse, K. Birman, and W. Vogels, "Astrolabe: A robust and scalable technology for distributed system monitoring, management, and data mining," *ACM Trans. Comput. Syst.*, vol. 21, no. 2, pp. 164–206, 2003.

[2] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd ed. Burlington, MA, USA: Morgan Kaufmann, 2006.

[3] C. Zhang and S. Zhang, "Association rule mining: Models and algorithms," in *Lecture Notes Computer Science* (LNAI 2307). Berlin/Heidelberg, Germany: Springer-Verlag, 2002.

[4] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in *Proc. SIGMOD*, 1993, pp. 207–216.

[5] R. Agrawal and R. Srikant, "Fast algorithms formining association rules," in *Proc. Int. Conf. Large Data Bases*, 1994, pp. 487–499.

[6] R. Srikant and R. Agrawal, "Mining quantitative association rules in large relational tables," in *Proc. ACM SIGMOD*, 1996, pp. 1–12.

[7] K. Sun and B. Fengshan, "Mining weighted association rules without preassigned weights," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 4, pp. 489–495, Apr. 2008.

[8] C. Silverstein, S. Brin, and R. Motwani, "Beyond market baskets: Generalizing association rules to dependence rules," *Data Mining Knowl. Discovery*, vol. 2, no. 1, pp. 39–68, 1998.

[9] B. Alatas and E. Akin, "An efficient genetic algorithm for automated mining of both positive and negative quantitative association rules," *Soft Comput.*, vol. 10, no. 3, pp. 230–237, 2006.

[10] S. Brin, R. Motwani, and C. Silverstein, "Beyond market baskets: Generalizing association rules to correlations," in *Proc. ACM SIGMOD Conf.*, 1997, pp. 265–276.

[11] D. Taniar, W. Rahayu, and O. Daly, "Mining hierarchical negative association rules," *Int. J. Comput. Intell. Syst.*, vol. 5, no. 3, pp. 434–451, 2012.

[12] X. Wu, C. Zhang, and S. Zhang, "Efficient mining of both positive and negative association rules," *ACM Trans. Inf. Syst.*, vol. 22, no. 3, pp. 381–405, 2004.

[13] A. Eiben and J. Smith, *Introduction to Evolutionary Computing*. Berlin, Germany: Springer-Verlag, 2003.

[14] J. Mata, J. Alvarez, and J. Riquelme, "Mining numeric association rules with genetic algorithms," in *Proc. 5th Int. Conf. Artif. Neural Netw. Genetic Algorithms*, Apr. 2001, pp. 264–267.

[15] X. Yan, C. Zhang, and S. Zhang, "Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support," *Expert Syst. Appl.*, vol. 36, no. 2, pp. 3066–3076, 2009.

[16] J. Alcala-Fdez, N. Flugy-Pape, A. Bonarini, and F. Herrera, "Analysis of the effectiveness of the genetic algorithms based on extraction of association rules," *Fund. Inf.*, vol. 98, no. 1, pp. 1001–1014, 2010.

[17] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA, USA/White Plains, NY, USA: Addison-Wesley/Longman, 1989.

[18] B. Alatas and E. Akin, "MODENAR: Multi-objective diferential evolution algorithm for mining numeric association rules," *Appl. Soft Comput.*, vol. 8, no. 1, p. 646, 2008.

[19] A. Ghosh and B. Nath, "Multi-objective rule mining using genetic algorithms," *Inf. Sci.*, vol. 163, nos. 1–3, pp. 123–133, 2004.

[20] C. Coello, G. Lamont, and D. V. Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*. Norwell, MA, USA: Kluwer Academic, 2002.

[21] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Norwell, MA, USA: Kluwer Academic, 2001.

[22] D. Martin, A. Rosete, J. Alcala-Fdez, and F. Herrera, "A multi-objective evolutionary algorithm for mining quantitative association rules," in *Proc. 11th Int. Conf. Intell. Syst. Design Applicat.*, Nov. 2011, pp. 1397–1402.

[23] H. Qodmanan, M. Nasiri, and B. Minaei-Bidgoli, "Multi-objective association rule mining with genetic algorithm without specifying minimum support and minimum confidence," *Expert Syst. Applicat.*, vol. 38, no. 1, pp. 288–298, 2011.

[24] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.

[25] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009.

[26] J. Mata, J. Alvarez, and J. Riquelme, "An evolutionary algorithm to discover numeric association rules," in *Proc. ACM Symp. Appl. Comput.*, Mar. 2002, pp. 590–594.

[27] C. Borgelt, "Efficient implementations of Apriori and Eclat," in *Proc. Workshop Freq. Itemset Mining Implement.*, vol. 90. 2003, pp. 280–296.

[28] M. Zaki, "Scalable algorithms for association mining," *IEEE Trans. Knowl. Data Eng.*, vol. 12, no. 3, pp. 372–390, May–Jun. 2000.

[29] K. Deb, S. Agrawal, A. Pratab, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[30] Q. Chen and Y. Chen, "Discovery of structural and functional features in RNA pseudoknots," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 7, pp. 974–984, Jun. 2009.

[31] S. Chattopadhyay, S. Rakesh, L. Land, and U. Acharya, "Studying infant mortality rate: A data mining approach," *Health Technol.*, vol. 1, no. 1, pp. 25–34, 2011.

[32] F. Berzal, I. Blanco, D. Sanchez, and M. Vila, "Measuring the accuracy and interest of association rules: A new framework," *Intell. Data Anal.*, vol. 6, no. 3, pp. 221–235, 2002.

[33] S. Brin, R. Motwani, J. Ullman, and S. Tsur, "Dynamic itemset counting and implication rules for market basket data," *ACM SIGMOD Rec.*, vol. 26, no. 2, pp. 255–264, 1997.

[34] L. Geng and H. Hamilton, "Interestingness measures for data mining: A survey," *ACM Comput. Surveys*, vol. 38, no. 3, pp. 1–32, 2006.

[35] G. Piatetsky-Shapiro, "Discovery, analysis and presentation of strong rules," in *Knowledge Discovery in Databases*. Cambridge, MA, USA: MIT Press, 1991, pp. 229–248.

[36] S. Ramaswamy, S. Mahajan, and A. Silberschatz, "On the discovery of interesting patterns in association rules," in *Proc. 24th Int. Conf. Very Large Data Bases*, 1998, pp. 368–379.

[37] E. Shortliffe and B. Buchanan, "A model of inexact reasoning in medicine," *Math. Biosci.*, vol. 23, nos. 3–4, pp. 351–379, 1975.

[38] K.-I. Ahn and J.-Y. Kim, "Efficient mining of frequent itemsets and a measure of interest for association rule mining," *J. Inf. Knowl. Manage.*, vol. 3, no. 3, pp. 245–257, 2004.

[39] P. Tan, V. Kumar, and J. Srivastava, "Selecting the right interestingness measure for association patterns," in *Proc. 8th Int. Conf. KDD*, 2002, pp. 32–41.

[40] M. Fidelis, H. Lopes, and A. Freitas, "Discovering comprehensible classification rules with a genetic algorithm," in *Proc. Congr. Evol. Comput.*, 2000, pp. 805–810.

[41] K. Miettinen, *Nonlinear Multiobjective Optimization*. Norwell, MA, USA: Kluwer, 1999.

[42] J. Alcala-Fdez, A. Fernandez, J. Luego, J. Derrac, S. Garcia, L. Sanchez *et al.*, "Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework." *J. Multiple-Valued Logic Soft Comput.*, vol. 17, nos. 2–3, pp. 255–287, 2011.

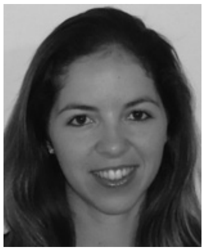
[43] J. Alcala-Fdez, L. Sanchez, S. Garcia, M. del Jesus, S. Ventura, J. O. J. Garrell *et al.*, "KEEL: A software tool to assess evolutionary algorithms to data mining problems," *Soft Comput.*, vol. 13, no. 3, pp. 307–318, 2009.

[44] H. Liu, F. Hussain, C. Tan, and M. Dash, "Discretization: An enabling technique," *Data Mining Knowl. Discovery*, vol. 6, no. 4, pp. 393–423, 2002.

[45] C.-H. Lee, "A Hellinger-based discretization method for numeric attributes in classification learning," *Knowl. Based Syst.*, vol. 20, no. 4, pp. 419–425, 2007.

[46] C. Tsai, C. Lee, and W. Yang, "A discretization algorithm based on class-attribute contingency coefficient," *Inf. Sci.*, vol. 178, no. 3, pp. 714–731, 2008.

[47] D. Janssens, T. Brijs, K. Vanhoof, and G. Wets, "Evaluating the performance of cost-based discretization versus entropy and error-based discretization," *Comput. Oper. Res.*, vol. 33, no. 11, pp. 3107–3123, 2006.



Diana Martín received the M.Sc. degree in applied informatics from Higher Polytechnic Institute José Antonio Echeverría (CUJAE), La Habana, Cuba, in 2010. She is currently pursuing the Ph.D. degree at the Soft Computing and Intelligent Information Systems Research Group, Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain.

She is currently an Assistant Professor with the Department of Artificial Intelligence and Infrastructure of Informatics Systems, CUJAE. Her research interests include data mining, association rules, knowledge extraction based on metaheuristics, and fuzzy systems.

Alejandro Rosete received the M.Sc. degree in applied informatics and the Ph.D. degree in informatics from Higher Polytechnic Institute José Antonio Echeverría (CUJAE), La Habana, Cuba, in 1995 and 2000, respectively.

He is the Head of the Department of Artificial Intelligence and Infrastructure of Informatics Systems, CUJAE. He has published over 40 papers. He is a co-author of the book *Lógica y Algoritmos* (Editorial Félix Varela, Habana, 2004). His research interests include metaheuristics, agent-oriented software engineering, decision making, data mining, fuzzy systems, and knowledge extraction based on metaheuristics.

Jesús Alcalá-Fdez (M'12) received the M.Sc. and Ph.D. degrees in computer science from University of Granada, Granada, Spain, in 2002 and 2006, respectively.

He is an Associate Professor with the Department of Computer Science and Artificial Intelligence, University of Granada, where he is a member of the Soft Computing and Intelligent Information Systems Research Group. He has published over 60 papers in international journals, book chapters, and conferences.

His research interests include data mining, knowledge extraction based on evolutionary algorithms, fuzzy rule-based systems, genetic fuzzy systems, multiobjective evolutionary algorithms, and data mining software.

He serves as a member of the Editorial Board of the *International Journal on Advances in Information Sciences and Service Sciences* and the *Scientific World Journal*. He has been the Chair of the Software Fuzzy Systems Task Force and a member of the Fuzzy Systems Technical Committee at the IEEE Computational Intelligence Society since 2011.

Francisco Herrera (M'10) received the M.Sc. and Ph.D. degree in mathematics from University of Granada, Granada, Spain, in 1988 and 1991, respectively.

He is a Professor with the Department of Computer Science and Artificial Intelligence, University of Granada. He has published over 250 papers. His research interests include computing with words and decision making, bibliometrics, data mining, data preparation, fuzzy rule based systems, genetic fuzzy systems, imbalanced classification, memetic algorithms and genetic algorithms, biometrics, cloud computing, and big data.

Dr. Herrera is the Editor-in-Chief of the international journal *Progress in Artificial Intelligence* (Springer). He is an Area Editor of *International Journal of Computational Intelligence Systems* and an Associate Editor of the journals IEEE TRANSACTIONS ON FUZZY SYSTEMS, *Information Sciences*, and *Knowledge and Information Systems*. He received the following honors and awards: ECCAI Fellow 2009, IFSA Fellow 2013, 2010 Spanish National Award on Computer Science ARITMEL to the Spanish Engineer on Computer Science, International Cajastur Mamdani Prize for Soft Computing (4th ed., 2010), IEEE TRANSACTIONS ON FUZZY SYSTEMS Outstanding 2008 Paper Award bestowed in 2011, 2011 Lotfi A. Zadeh Prize, Best Paper Award of the International Fuzzy Systems Association, and 2013 AEPIA Award to a scientific career in artificial intelligence.