Evolutionary Fuzzy Rule Induction Process for Subgroup Discovery: A Case Study in Marketing

María José del Jesus, Pedro González, Francisco Herrera, and Mikel Mesonero

***Abstract*—This paper presents a genetic fuzzy system for the data mining task of subgroup discovery, the subgroup discovery iterative genetic algorithm (SDIGA), which obtains fuzzy rules for subgroup discovery in disjunctive normal form. This kind of fuzzy rule allows us to represent knowledge about patterns of interest in an explanatory and understandable form that can be used by the expert. Experimental evaluation of the algorithm and a comparison with other subgroup discovery algorithms show the validity of the proposal. SDIGA is applied to a market problem studied in the University of Mondragón, Spain, in which it is necessary to extract automatically relevant and interesting information that helps to improve fair planning policies. The application of SDIGA to this problem allows us to obtain novel and valuable knowledge for experts.**

***Index Terms*—Data mining, descriptive induction, evolutionary algorithms, genetic fuzzy systems, subgroup discovery.**

I. Introduction

RULE learning is an important form of *predictive* machine learning, aimed at inducing a set of rules to be used for classification and/or prediction [1], [2]. Developments in *descriptive induction* have recently also attracted much attention from researchers interested in rule learning. The objective of *descriptive machine learning* is to discover of individual rules that define interesting patterns in data, and it includes approaches for mining association rules [3], for subgroup discovery [4], [5] and other nonclassificatory induction approaches such as clausal discovery [6] or database dependency [7] among others.

Subgroup discovery is a form of supervised inductive learning, which is defined as follows [4], [5]: given a population of individuals and a specific property of individuals in which we are interested, find population subgroups that are statistically "most interesting," e.g., are as large as possible and have the most unusual distributional characteristics with respect to the property of interest. The concept was initially formulated by Klösgen in EXPLORA [4] and by Wrobel in MIDOS [5]. Later, subgroup discovery has been applied in different fields: in medicine, in problems such as coronary heart disease risk group detection [8] or the extraction of comprehensible models for gene expression data sets [9]; in marketing, in problems such as decision support in a direct mailing campaign and in a public advertising campaign [10]; or in problems with multiple sources of information useful for the expert, as in the analysis of traffic data [11].

It must be noted that subgroup discovery aims to discover individual rules (or local patterns of interest, very frequent—hence typical—or very rare—hence atypical), which must be represented in an explicit symbolic form and must be relatively simple in order to be recognized as actionable by potential users. Therefore, the subgroups discovered in data are of a more explanatory nature, and the interpretability of the extracted knowledge for the final user is a crucial aspect in this field.

As claimed by Dubois *et al.* in [12], the use of fuzzy sets to describe associations between data extends the types of relationships that may be represented, facilitates the interpretation of rules in linguistic terms, and avoids unnatural boundaries in the partitioning of the attribute domains. It is especially useful in medical, control, or economic fields where the boundaries of a piece of information used may not be clearly defined. In fact, the use of linguistic variables and linguistic terms in a machine learning process has been thoroughly explored by various authors in predictive induction (see, for instance, Ishibuchi *et al.* [13] for a complete and understandable up-to-date description on the design of classification and modelling fuzzy systems). In descriptive induction, there are some proposals specially for the extraction of fuzzy association rules [14], [15], but to the best of our knowledge no proposals have been made in the subgroup discovery area.

A fuzzy approach for a subgroup discovery process, which considers linguistic variables with linguistic terms in descriptive fuzzy rules, allows us to obtain knowledge in a similar way to human reasoning. In order to understand this it is enough to consider that much of the logic behind human reasoning is not traditional two-valued or even multivalued logic but logic with fuzzy truths, fuzzy connectives, and fuzzy rules of inference. Fuzzy rules are naturally inclined towards coping with linguistic knowledge, thereby producing more interpretable and actionable solutions in the field of subgroup discovery and in general in the analysis of data to establish relationships and identify patterns [16].

In the specialized bibliography, there are different approaches for the extraction of fuzzy rules: decision trees [17], [18], artificial neural networks [19], and genetic algorithms (GAs) [20], among others.

GAs are search algorithms based on natural genetics that provide robust search capabilities in complex spaces [21]. Although

Manuscript received November 28, 2005; revised May 9, 2006 and June 20, 2006. This work was supported in part by the Spanish Ministry of Education and Science under Projects TIC-2002-04037-C05-01, TIC-2002-04037-C05-04, TIN-2005-08386-C05-03, and TIN-2005-08386-C05-01.

M. J. Del Jesus and P. González are with the Department of Computer Science, University of Jaén, 23071 Jaén, Spain (e-mail: mjjesus@ujaen.es; pglez@ujaen.es).

F. Herrera is with the Department of Computer Science and Artificial Intelligence, University of Granada, 18071 Granada, Spain (e-mail: herrera@decsai.ugr.es).

M. Mesonero is with the Department of Marketing, University of Mondragón, 20500 Mondragón, Spain (e-mail: mmesoner@eteo.mondragon.edu).

Digital Object Identifier 10.1109/TFUZZ.2006.890662

they were not specifically designed for learning, they are widely used for evolving rules and for pattern association in data mining and knowledge discovery [22], [23].

The hybridization between fuzzy logic and GAs, called genetic fuzzy systems (GFSs) [20], has attracted considerable attention in the Computational Intelligence community. GFSs provide novel and useful tools for pattern analysis and for extracting new kinds of useful information. The interpretability of GFSs in terms of fuzzy if-then rules provides the main advantage over other techniques.

This paper describes a new GFS proposal for subgroup discovery called the subgroup discovery iterative genetic algorithm (SDIGA). This approach allows us to obtain a set of understandable fuzzy rules with a flexible structure that describes different subgroups in data.

SDIGA is applied to an interesting problem in the field of marketing: the study of the influence that planning variables of a trade fair has on the successful achievement of its objectives. This real-world problem is relevant because nowadays, face to face contact with the clients continues to be fundamental in the development of marketing actions, and trade fairs are, in this sense, a basic instrument in company marketing policies, especially in industrial marketing. Due to the high investment in terms of both time and money, the extraction of relevant and interesting information that helps to improve planning policies for fairs is necessary.

The paper is arranged as follows. In Section II, some preliminaries are described: the definition for the subgroup discovery task, the kind of fuzzy rules and quality measures used, the main proposals in the specialized bibliography for subgroup discovery systems, and a brief overview of GFSs for rule induction. The evolutionary approach to obtain subgroup discovery descriptive fuzzy rules is explained in Section III. In Section IV, the results obtained in the market problem are analyzed. Finally, in Section V, the conclusions and further research are outlined.

II. Preliminaries: Subgroup Discovery

In this section, we will briefly describe the subgroup discovery task, the kind of fuzzy rule used in this proposal, the quality measures considered to evaluate a single rule and/or a set of rules, the main approaches in the specialized bibliography, and some considerations about the use of GFSs in rule induction processes.

A. Introduction to Subgroup Discovery

Within the descriptive machine learning area, subgroup discovery has recently received a great deal of attention from researchers. It represents a form of supervised inductive learning in which, given a set of data and a property of interest to the user (target variable), an attempt is made to locate subgroups that are statistically "most interesting" for the user. In this sense, a subgroup is interesting if it has an unusual statistical distribution with respect to the property of interest. Descriptive machine learning methods for subgroup discovery have the objective of discovering interesting properties of subgroups by obtaining *simple* rules (i.e., with an understandable structure and with few variables), which are *highly significant* and *with high support* (i.e., covering many of the instances of the target class).

An induced subgroup description has the form of an implication

$$\text{Cond} \rightarrow \text{Class} \tag{1}$$

where the property of interest for subgroup discovery is the class value Class that appears in the consequent part of the rule and the antecedent part of the rule Cond is a conjunction of features (attribute-value pairs) selected from the features describing the training instances.

Subgroup discovery is usually seen as different from classification, as it addresses different goals. The goal of classification rule learning is to generate models consisting of sets of rules describing class characteristics of all the training examples, trying to maximize the classification accuracy of the induced set of rules. In contrast, subgroup discovery aims to discover individual rules of interest, which must be represented in explicit symbolic form and must be relatively simple in order to discover interesting population subgroups. In addition, the set of individual rules obtained by the subgroup discovery task will not necessarily describe all the examples.

The subgroup discovery task relies on the following main properties.

- The description language specifying the subgroups must be adequate to be applied effectively by the potential users. The subgroup description consists of a set of expressions. In the simplest case, each expression is single-valued; however, negation or internal disjunctions are also possible.
- The quality function measuring the interest of the subgroup. A variety of quality functions have been proposed (see [4], [24], and [8], for instance). The applicable set of quality functions is determined by the type of the target variable, the type of rule, and the problem considered. In the next section, we will describe several quality measures used in subgroup discovery algorithms.
- The search strategy is very important, since the dimension of the search space has an exponential relation to the number of features (or variables) and values considered.

In this proposal, we use fuzzy rules in disjunctive normal form (DNF fuzzy rules) as description language to specify the subgroups, which permit a disjunction for the values of any variable present in the antecedent part. Bellow, the notation used in this paper is described. We consider a problem with the following.

- A set of features

$$\{X_m / m = 1, \ldots, n_v\} \tag{2}$$

  used to describe the subgroups, where n_v is the number of features. These variables can be categorical or numerical.
- A set of values for the target variable

$$\{\text{Class}_j / j = 1, \ldots, n_c\} \tag{3}$$

  where n_c is the number of values for the target variable considered.

- A set of examples

$$\left\{E^k = (e_1^k, e_2^k, \ldots, e_{n_v}^k, \text{Class}_j)/k = 1, \ldots, n_s\right\} \quad (4)$$

where Class_j is the target variable value for the sample E^k (i.e., the class for this example) and n_s is the number of examples for the descriptive induction process.

- A set of linguistic labels for the numerical variables. The number of linguistic labels and the definition for the corresponding fuzzy sets depend on each variable

$$X_m : \left\{\text{LL}_m^1, \text{LL}_m^2, \ldots, \text{LL}_m^{l_m}\right\}. \quad (5)$$

In this expression, we represent the set of linguistic labels for the variable X_m, which has l_m different linguistic labels to describe its domain in an understandable way.

Then, a fuzzy rule R^i can be described as

$$R^i : \text{Cond}^i \rightarrow \text{Class}_j \quad (6)$$

where the antecedent describes the subgroup in disjunctive normal form. The DNF fuzzy rule can be expressed as

$$R^1 : \text{ If } X_1 \text{ is } \text{LL}_1^1 \text{ or } \text{LL}_1^3 \text{ and } X_7 \text{ is } \text{LL}_7^1 \text{ then Class}_j. \quad (7)$$

It must be noted that any subset of the complete set of variables (with any combination of linguistic labels related to the operator OR) can take part in the rule antecedent. In this way a subgroup is a compact and interpretable description of patterns of interest in data.

For this kind of fuzzy rule, we consider the following.

- An example E^k *verifies the antecedent part of a rule* R^i if

$$\text{APC}(E^k, R^i) = \text{T}\left(\text{TC}\left(\mu_{\text{LL}_1^1}\left(e_1^k\right), \ldots, \mu_{\text{LL}_1^{l_1}}\left(e_1^k\right)\right), \ldots, \text{TC}\left(\mu_{\text{LL}_{n_v}^1}(e_{n_v}^k), \ldots, \mu_{\text{LL}_{n_v}^{l_{n_v}}}\left(e_{n_v}^k\right)\right)\right) > 0 \quad (8)$$

where:

- antecedent part compatibility (APC) is the degree of compatibility between an example and the antecedent part of a fuzzy rule, i.e., the degree of membership for the example to the fuzzy subspace delimited by the antecedent part of the rule;
- $\text{LL}_{n_v}^{l_{n_v}}$ is the linguistic label number l_{n_v} of the variable n_v;
- $\mu_{\text{LL}_{n_v}^{l_{n_v}}}(e_{n_v}^k)$ is the degree of membership for the value of the feature n_v for the example E^s to the fuzzy set corresponding to the linguistic label l_{n_v} for this variable (n_v);
- T is the t-norm selected to represent the meaning of the AND operator—the fuzzy intersection—in our case the minimum t-norm;
- TC is the t-conorm selected to represent the meaning of the OR operator—the fuzzy union—which in our case is the maximum t-conorm.

- An example E^k is *covered by a rule* R^i if

$$\text{APC}(E^k, R^i) > 0 \quad \text{AND} \quad E^k \in \text{Class}_j. \quad (9)$$

This means that an example is covered by a rule if the example has a degree of membership higher than zero to the fuzzy input subspace delimited by the antecedent part of the fuzzy rule, and the value indicated in the consequent part of the rule agrees with the value of the target feature for the example. For the categorical variables, the degrees of membership are zero or one.

B. Quality Measures in Subgroup Discovery

One of the most important aspects of any subgroup discovery algorithm—and a determining factor in the quality of the approach—is the quality measure to be used, both to select the rules and to evaluate the results of the process.

To solve the subgroup discovery tasks, objective and subjective quality measures can be considered. For automatic rule induction, only objective quality criteria can be applied. However, to evaluate the quality of descriptions of induced subgroups and their use in decision making, subjective criteria are important, although they are more difficult to evaluate and only used in a final analysis of the extracted knowledge.

Some of the *subjective* interest measures are *usefulness* [4], *actionability* [25], [26], *operationality* [10], *unexpectedness* [26], *novelty* [4], and *redundancy* [4].

Objective measures for descriptive induction evaluate each subgroup individually but can be complemented by their variants to compute the mean of the induced set of descriptions of subgroups, allowing comparison between different subgroup discovery algorithms. There are different studies about objective quality measures for the descriptive induction process [24], [25], [27] but it is difficult to reach an agreement about their use. Below, the more widely used objective quality measures in the specialized bibliography of subgroup discovery are described.

- *Coverage of a rule* [28]*:* this measures the percentage of examples covered on average by one rule of the induced set of rules

$$\text{Cov}(R^i) = \text{Cov}(\text{Cond}^i \rightarrow \text{Class}_j) = p(\text{Cond}^i) = \frac{n(\text{Cond}^i)}{n_s} \quad (10)$$

where:

- $n(\text{Cond}^i)$ is the number of examples which verifies the condition Cond^i described in the antecedent (independently of the class to which belongs), i.e., is the number of examples which verify (8);
- n_s is the number of examples.

It must be noted that, in this expression, the coverage is computed by a count and not by a sum of membership degrees to the fuzzy area delimited by the antecedent.

The *average coverage for the set of rules* finally obtained is calculated by the following expression

$$\text{COV} = \frac{1}{n_r}\sum_{i=1}^{n_r}\text{Cov}(R^i) \quad (11)$$

where n_r is the number of induced rules.

- *Support of a rule:* In descriptive induction processes, support for a rule is a standard measure that considers, by means of an expression that can vary in different proposals, the number of examples satisfying both the antecedent and the consequent parts of the rule. Lavrac *et al.* compute in [28] the overall support as the percentage of target examples (positive examples) covered by the rules. The support of a rule is defined as the frequency of correctly classified examples covered

$$\begin{aligned}\text{Sup}_1(R^i) &= \text{Sup}_1(\text{Cond}^i \rightarrow \text{Class}_j) \\ &= p(\text{Class}_j.\text{Cond}^i) = \frac{n(\text{Class}_j \cdot \text{Cond}^i)}{n_s}\end{aligned} \tag{12}$$

  where $n(\text{Class}_j.\text{Cond}^i)$ is the number of examples that satisfy the conditions for the antecedent (Cond^i) and simultaneously belong to the value for the target variable (Class_j) indicated in the consequent part of the rule. In other words, $n(\text{Class}_j \cdot \text{Cond}^i)$ is the number of examples verifying (9). The support for a set of rules is computed by

$$\text{SUP} = \frac{1}{n_s}\sum_{j=1}^{n_c} n(\text{Class}_j \cdot \bigvee_{\text{Cond}^i \rightarrow \text{Class}_j} \text{Cond}^i). \tag{13}$$

  It must be noted that in this expression the examples that belong to many rules are considered only once and, as in the coverage measure, the degree of membership (for numerical variables considered as linguistic variables) is not considered but only the count.
  In (12), the support of a rule is computed dividing by the total number of examples. It can also be computed in other ways, such as dividing by the number of examples of the class or other variations.
- *Size (for a set of rules):* The size of a set of rules is a complexity measure calculated as the number of induced rules (n_r). Complexity can also be measured as the mean number of obtained rules per class or the mean of variables per rule.
- *Significance of a rule* [4]*:* indicates the significance of a finding if measured by the likelihood ratio of a rule

$$\begin{aligned}\text{Sig}(R^i) &= \text{Sig}(\text{Cond}^i \rightarrow \text{Class}_j) \\ &= 2 \cdot \sum_{j=1}^{n_c} n(\text{Class}_j.\text{Cond}^i) \log \frac{n(\text{Class}_j.\text{Cond}^i)}{n(\text{Class}_j) \cdot p(\text{Cond}^i)}\end{aligned} \tag{14}$$

  where $p(\text{Cond}^i)$, computed as $n(\text{Cond}^i)/n_s$, is used as a normalized factor.
  It must be noted that, although each subgroup description (i.e., rule) is for a specific class value, the significance measures impartially the novelty in the distribution for all the class values.
  The significance for a set of rules is computed as follows:

$$\text{SIG} = \frac{1}{n_R}\sum_{i=1}^{n_R} \text{Sig}(R^i). \tag{15}$$

- *Unusualness of a rule:* It is defined as the *weighted relative accuracy* of a rule [29]

$$\begin{aligned}&\text{WRAcc}(\text{Cond}^i \rightarrow \text{Class}_j) \\ &\quad = \frac{n(\text{Cond}^i)}{n_s}\left(\frac{n(\text{Class}_j.\text{Cond}^i)}{n(\text{Cond}^i)} - \frac{n(\text{Class}_j)}{n_s}\right).\end{aligned} \tag{16}$$

  The weighted relative accuracy of a rule can be described as the balance between the coverage of the rule $(p(\text{Cond}^i))$ and its accuracy gain $(p(\text{Class}_j.\text{Cond}^i) - p(\text{Class}_j))$. It must be noted that the higher a rule's unusualness, the more relevant it is.
  The unusualness for a set of rules is computed as

$$\text{WRACC} = \frac{1}{n_R}\sum_{i=1}^{n_R} \text{WRAcc}(R^i). \tag{17}$$

  It must be noted that all the quality measures described here are crisp (nonfuzzy) measures because the proposals in the specialized bibliography of subgroup discovery do not consider fuzzy rules. These measures are used to compare the results of our proposal with other classic subgroup discovery algorithms.

The evolutionary algorithm described in this paper induces fuzzy rules guided by the following quality factors.

- *Confidence of a fuzzy rule:* The confidence of a rule is a standard measure that determines the relative frequency of examples satisfying the complete rule among those satisfying only the antecedent. It can be computed with different expressions proposed in the bibliography. In this paper, the expression used for confidence reflects the degree to which the examples within the zone of the space marked by the antecedent verify the information indicated in the consequent part of the rule. To calculate this factor, we use an adaptation of Quinlan's accuracy expression [30] in order to generate fuzzy classification rules [31]: the sum of the degree of membership of the examples of this class (the examples covered by this rule) to the fuzzy input subspace determined by the antecedent, divided by the sum of the degree of membership of all the examples that verify the antecedent part of this rule (irrespective of their class) to the same zone

$$\text{Conf}(\text{R}^{\text{i}}) = \frac{\sum_{E^k \in E / E^k \in \text{Class}_j} \text{APC}(E^k, R^i)}{\sum_{E^k \in E} \text{APC}(E^k, R^i)}. \tag{18}$$

- *Support of a fuzzy rule,* which in our proposal is defined as the degree of coverage that the rule offers to examples of that class, is computed as

$$\text{Sup}_2(R^i) = \frac{n(\text{Class}_j.\text{Cond}^i)}{n(\text{Class}_j)} \tag{19}$$

  where $n(\text{Class}_j)$ is the number of examples of the class j. A variation of this measure will be detailed in Section III.

C. Related Works in Subgroup Discovery

In the specialized bibliography, different methods have been developed that obtain descriptions of subgroups represented in

different ways and using different quality measures. Next we briefly describe some of them.

- The first approach developed for subgroup discovery was EXPLORA [4]. It uses decision trees for the extraction of rules. The rules are specified by defining a descriptive schema and implementing a statistical verification method. The interest of the rules is measured using measures such as evidence, generality, redundancy, and simplicity.
- MIDOS [5] applies the EXPLORA approach to multirelational databases. It uses optimistic estimation and minimum support pruning. The goal is to discover subgroups of the target relation (defined as first order conjunctions) that have unusual statistical distributions with respect to the complete population. The quality measure is a combination of unusualness and size.
- SubgroupMiner [32] is an extension of EXPLORA and MIDOS. It is an advanced subgroup discovery system that uses decision rules and interactive search in the space of the solutions, allowing the use of very large databases by means of the efficient integration of databases, multirelational hypotheses, visualization based on interaction options, and the discovery of structures of causal subgroups. This algorithm uses as its standard quality function the classical binomial test to verify if the statistical distribution of the target is significantly different in the extracted subgroup.
- SD [27] is a rule induction system guided by expert knowledge: instead of defining an optimal measure to search and select automatically the subgroups, the objective is to help the expert in performing flexible and effective searches on a wide range of optimal solutions.
- CN2-SD [28] (a modified version of the CN2 classification rule algorithm [1]) induces subgroups in the form of rules using as quality measure the relation between true positives and false positives. CN2-SD uses a modified weighted relative accuracy as the quality measure for rule selection. In this paper, we use this algorithm to compare with our proposal, and it is further described in Appendix A.
- Relational subgroup discovery (RSD) [33] has the objective of obtaining population subgroups that are as large as possible, with a statistical distribution as unusual as possible with respect to the property of interest, and are different enough to cover most of the target population. It is a recent upgrade of the CN2-SD algorithm that enables relational subgroup discovery.
- APRIORI-SD [34] is developed by adapting the APRIORI association rule learning algorithm [35] to subgroup discovery. To achieve this, APRIORI-C [36], a modification of the original APRIORI to learn classification rules, has been used, including a new postprocessing mechanism, a new quality measure for the induced rules (the weighted relative accuracy), and using probabilistic classification of the examples. For the evaluation of the set of rules, the area under the *receiver operating characteristic* (ROC) curve is used, in conjunction with the support and significance of each individual rule and the size and accuracy of the set of rules.
- Intensive Knowledge [37] uses several types of application background knowledge to improve the quality of the results of the subgroup discovery task and the efficiency of the search method.

As can be seen, there is increasing interest in the development of subgroup discovery algorithms from association rule-learning algorithms.

D. Genetic Fuzzy Systems for Rule Induction Processes

Fuzzy systems have shown their usefulness in solving a wide range of problems in different application domains. The use of GAs [21], [38] in the design fuzzy systems allows us to equip them with the learning and adaptation capabilities. The result of this hybridization between fuzzy logic and GAs leads to genetic fuzzy systems (GFSs) [20], [39]. A GFS is basically a fuzzy system enhanced by a learning process based on a GA.

Although GAs were not specifically designed for learning, but rather as global search algorithms, they offer a set of advantages for knowledge extraction and specifically for rule induction processes.

- They tend to cope well with attribute interaction because they usually evaluate a rule as a whole via a fitness function rather than evaluating the impact of adding/removing one condition to/from a rule.
- They have the ability to scour a search space thoroughly and to handle a fitness function adapted to the problem to be solved. The fitness function can contain different criteria such as the ability to penalize overlap among rules or sets of rules with too many rules or a problem-specific quality measure, among others.
- In addition, the genetic search performs implicit backtracking in its search of the rule space, thereby allowing it to find complex interactions that other nonbacktracking searches would miss.
- An additional advantage over other conventional rule-learning algorithms is that the search is carried out among a set of competing candidate rules or sets of rules.

However, this is not to say that GAs are inherently superior to rule induction algorithms, as no rule discovery algorithm is superior in all cases [2].

Since the early 1990s, GAs have been used for the design of fuzzy rule-based systems, but mainly with predictive aims, in control and pattern classification problems. Rule induction algorithms for subgroup discovery (the aim of which is fundamentally descriptive) share characteristics with algorithms that guide the induction process using predictive quality measures. In this section, we will describe some of the main GFS proposals for rule induction, no matter what their final aim is.

The genetic representation of solutions is the most determining aspect of any GFS proposal. In this sense, the proposals in the specialized literature follow two approaches in order to encode rules within a population of individuals [20].

- The "Chromosome = Rule" approach, in which each individual codifies a single rule.
- The "Chromosome =Set of rules" approach, also called the Pittsburgh approach, in which each individual represents a set of rules. Carse *et al.* [40] and Wang *et al.*'s [41] proposals are examples of GFSs that use this representation model.

In turn, within the "Chromosome = Rule" approach, three learning proposals can be found.

- The Michigan approach in which each individual codifies a single rule. They are rule-based systems, which use a GA and a reinforcement component to learn rules that guide its performance in a certain environment [42]. In [43]–[45] Michigan-style GFSs can be found.
- The iterative rule-learning (IRL) approach, in which each chromosome represents a rule but the GA solution is the best individual obtained and the global solution is formed by the best individuals obtained when the algorithm is run multiple times. SLAVE [46] and MOGUL [47] are GFSs of this type.
- The "cooperative-competitive" approach, in which the complete population or a subset of it codifies the rule base. In [48] and [23], we can find two genetic learning approaches following this idea.

In the extraction of rules for the subgroup discovery task, the "Chromosome = Rule" approach is more suited because the objective is to find a reduced set of rules in which the quality of each rule is evaluated independently of the rest, and it is not necessary to evaluate jointly the set of rules. This is the encoding approach used in the following evolutionary proposal.

III. SDIGA: Hybrid GA for the Induction of Subgroup Discovery Fuzzy Rules

In this section, an evolutionary model for the extraction of fuzzy rules for a subgroup discovery task, SDIGA, is presented. The objective is to obtain a set of rules that describe subgroups for all the values of the target feature. To ensure this, the SDIGA algorithm must be run once by each value of the target feature. That is to say, each run of SDIGA obtains a set of rules for a specific value of the target feature.

SDIGA follows the IRL approach, previously described, and works as follows.

- The core of SDIGA is a GA that uses a postprocessing step based on a simple local search, a hill-climbing procedure. The hybrid GA extracts one simple and interpretable fuzzy rule with an adequate level of support and confidence. The consequent part of the rule is composed of a single feature (the property of interest for the user, or target feature). The postprocessing step consists of a local search process increasing its generality.
- This hybrid GA is included in an iterative process for the extraction of a set of fuzzy rules for the description of subgroups supported by different areas (not necessarily disjuncts) of the instance space. We obtain a set of solutions generated in successive runs of the GA corresponding to the same value of the target feature.

The model uses fuzzy rules in DNF format. DNF fuzzy rules offer a more flexible structure to the rules, allowing each variable to take more than one value and facilitating the extraction of more general rules. In this kind of fuzzy rule, as defined in (7), fuzzy logic contributes to the interpretability of the extracted rules owing to the use of a knowledge representation close to the expert, also allowing the use of numerical features without previous discretization. The fuzzy sets corresponding to the linguistic labels $(\mathrm{LL}_m^1, \ldots, \mathrm{LL}_m^{l_m})$ are defined by means of the corresponding membership functions, which can be specified by the user or defined by means of a uniform partition if the expert knowledge is not available. In this algorithm, we use uniform partitions with triangular membership functions, as shown in Fig. 1 for a variable with five linguistic labels. The proposal can be used, as previously mentioned, with a predefined set of linguistic labels (and the corresponding fuzzy sets). This fuzzy partition can be defined by a heuristic approach that places the fuzzy sets in such a way that each of them will cover approximately the same number of data, if the expert so desires. But it must be considered that, depending on the problem, the interpretation of the resulting fuzzy rules could be decreased. Moreover, if it is necessary, a preliminary data analysis that detects outliers in data can be carried out before the determination of the fuzzy partitions. In this way, a specific analysis of them can be realized and the fuzzy partition (without these outlier data) is not biased by them.

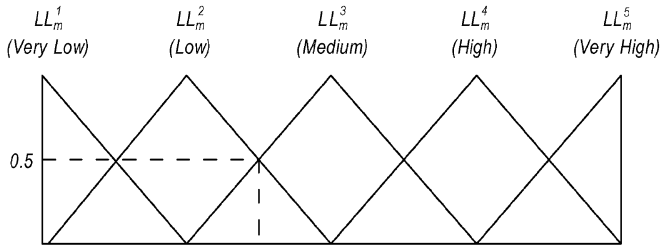

Fig. 1. Example of fuzzy partition for a numerical variable.

It must be noted that the hybrid GA is included in an iterative process for the extraction of *different* rules in successive runs of the algorithm. For this purpose, when a run of the hybrid GA has finished and a fuzzy rule has been obtained, the positive instances of the rule (covered examples) are marked to prevent the obtaining of a new rule that covers exactly the same examples in the following GA runs. At the first iteration, none of the instances is covered, because there are no extracted rules. This method to guide the GA evolution over different—although maybe overlapping—fuzzy rules is explained in detail in the next section.

This is an outline of the basis of the model. Below, we describe in detail the GA and the iterative rule extraction model. The results of a comparison of the proposal with other subgroup discovery algorithms are also detailed.

A. Hybrid Genetic Algorithm for the Induction of a Fuzzy Rule

The hybrid GA extracts a single DNF fuzzy rule in an attempt to optimize the confidence and support. Next, we describe the elements of the hybrid GA: the chromosome representation, the fitness function, the reproduction model, and the postprocessing phase of the hybrid GA.

1) Chromosome Representation: In a subgroup discovery task, we have a number of descriptive features and a single target feature of interest (describing the subgroups).

The GA discovers a DNF fuzzy rule whose consequent is prefixed to one of the possible values of the target feature. So, in this proposal, each candidate solution is coded according to the

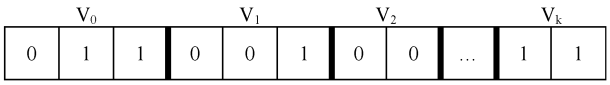

Fig. 2. Encoding model of a rule.

"Chromosome = Rule" approach. Only the antecedent is represented in the chromosome, and all the individuals in the population are associated with the same value of the target feature. As we have mentioned above, this form of categorizing the target feature means that the evolutionary algorithm must be run many times in order to discover rules of different classes.

All the information relating to a rule is contained in a fixed-length chromosome with a binary representation in which, for each feature, a bit for each one of the possible values of the feature is stored. In this way, if the corresponding bit contains the value 0, it indicates that the value is not used in the rule; and if the bit contains the value 1, it indicates that the corresponding value is included. If a rule contains all the bits corresponding to a feature with the value 1, or all of them contain the value 0, this indicates that this feature has no relevance for the information contributed in the rule, and so this feature is ignored. In these cases, the feature does not take part in the rule.

This takes us to a binary representation model with as many genes by variable as possible values for the same one, as can be seen in Fig. 2. In this figure, V_0 and V_1 have three possible values and V_2 and V_k have two possible values. In this example, neither V_2 nor V_k take part in the rule (V_2 does not take any of its values and V_k takes all, and so both variables are irrelevant for the rule).

The set of possible values for the categorical features is that indicated by the problem, and for numerical variables it is the set of linguistic terms determined heuristically or with expert information.

2) Fitness Function: In this process of rule discovery, the objective is to obtain rules with high confidence, and that are understandable and general. It means that the problem has at least two objectives to maximize: the support and the confidence of the rule. To achieve this, the weighted sum method that weights a set of objectives into a single objective is the simplest approach and lets us introduce the expert criteria related to the importance of the objectives for a specific problem in the rule generation process. This method has one difficulty: the determination of proper values for the weights, which depends on the importance of each objective in the context of the problem. The weight of one objective is chosen in proportion to the objective's relative importance in the problem. So, this proposal uses a weighted lineal combination in the following way:

$$\text{fitness}(c) = \frac{\omega_1 \times \text{Sup}_3(c) + \omega_2 \times \text{Conf}(c)}{\omega_1 + \omega_2} \tag{20}$$

where confidence (Conf) and support (Sup_3) of the rule are defined as follows.

- *Confidence:* This determines the accuracy of the rule, in that it reflects the degree to which the examples within the zone of the space determined by the antecedent verify the information specified in the consequent of the rule, and it is computed as in (18).
- *Support:* This measures the degree of coverage that the rule offers to examples belonging to the class specified in the rule consequent. It is calculated in a different way than in (14) to promote different fuzzy rules being obtained in different runs of the hybrid GA. To do so, for the computation of the support, we only consider the examples not marked (i.e., the examples not covered by other fuzzy rules previously obtained by means of the past runs of the hybrid GA). This strategy helps to reach the objective of the proposal, the extraction of useful knowledge from different examples of the complete dataset. Thus, the support is defined as the quotient between the examples of this partial set covered by the rule represented in the chromosome and the total number of examples of this partial set

$$\text{Sup}_3(R^i) = \frac{\text{Ne}^+(R^i)}{\text{Ne}_{\text{NC}}} \tag{21}$$

where:
  - Ne_{NC} is the number of examples left uncovered by the previous rules;
  - $\text{Ne}^+(R^i)$ is the number of examples covered by the rule that are left uncovered by the previous rules.

  Again, we use (9) to determine when an example is covered by a rule.

  This way of measuring support is sensible, when using the GA within an iterative process, in order to obtain different rules each time the GA is run. From the second iteration, rules that cover examples belonging to zones delimited by previously obtained rules are penalized because the support factor only considers examples that have not been described by rules already obtained. No distance function is used, as differences are penalized on a phenotypical level.

The overall objective of the evaluation function is to direct the search towards rules that maximize accuracy, minimizing the number of negative examples and examples not covered.

3) Reproduction Model and Genetic Operators: The GA uses a steady-state reproduction model [38], in which the original population is only modified through the substitution of the worst individuals by individuals resulting from crossover and mutation. The recombination is carried out by means of a two-point crossover operator and a biased random mutation operator.

The crossover is applied over the two best individuals of the population, obtaining two new individuals, which will substitute the two worst individuals in the population. This strategy leads to a high selective pressure with the aim of getting a quick convergence of the algorithm, due to our applying of the GA iteratively to get different rules and also applying a local search to improve locally each rule. Therefore, we do not need to introduce high diversity in the GA search by means of the parent selection mechanism; we prefer to have a quick convergence.

Mutation is carried out as follows. First, according to the mutation probability, the chromosome and the gene of the chromosome to be mutated are determined. Then, the biased random mutation operator is applied in two different ways, with probability 0.5 in each case. In the first way, the mutation causes the elimination of the variable to which the gene corresponds, setting to 0 all the values of this variable, as shown in Fig. 3. The second type of mutation randomly assigns 0 or 1 to all the values

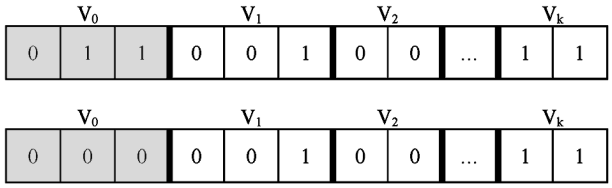

Fig. 3. Mutation type 1: elimination of the variable V_0.

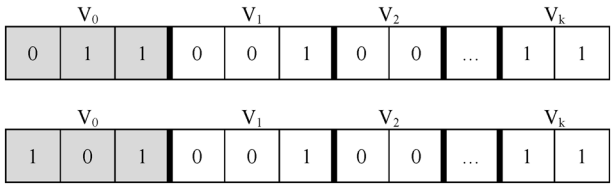

Fig. 4. Mutation type 2: random setting for variable V_0.

```
START
  Best_Rule ← R
  Best_support ← support(R)
  Better ← True
  REPEAT WHILE Better
    Better ← False
    FOR (m=1 to n_v)
      R'_m ← Best_Rule without considering variable m
      IF (support(R'_m)>=support(R) AND confidence(R'_m)>=confidence(R))
        Better ← True
        IF (support (R'_m) > Best_support)
          Best_support ← support (R'_m)
          Best_Rule ← R'_m
    END FOR
  END WHILE
  IF (confidence(Best_Rule)>=min_conf)
    Return Best_Rule
  ELSE
    Return R
END
```

Fig. 5. The postprocessing phase of the hybrid GA.

of the variable, as can be seen in Fig. 4. So, half the mutations have the effect of eliminating the corresponding variable, and the rest randomly set the values for the variable to be muted.

4) Hybrid GA Postprocessing Phase: Local Search Algorithm: The postprocessing phase, which improves the obtained rule by a hill-climbing process, modifies the rule in order to increase the degree of support. To accomplish this, in each iteration a variable is selected such that when it is eliminated, the support of the resulting rule is increased; in this way more general rules are obtained. Finally, the optimized rule will substitute the original one only if it overcomes minimum confidence. The diagram of the postprocessing phase is as shown in Fig. 5.

B. Iterative Rule Extraction Model

The fuzzy descriptive rule extraction model follows the IRL approach, and its objective is to obtain a set of rules giving information on the majority of available examples for each value of the target feature.

The data mining process is carried out by means of an iterative algorithm allowing the generation of several rules (one for each GA run). All the rules extracted within this iterative algorithm correspond to the same value of the target feature. The iterative algorithm continues obtaining rules, whereas the generated rules:

- reach a minimum level of confidence (previously specified);
- give information on areas of the search space in which there are examples not described by the rules generated in previous iterations.

The iterative process promotes the generation of different rules (in the sense that they give information on different groups of examples). This is achieved by penalizing—once a rule is obtained—the set of examples represented by the same one in order to generate future rules.

It is important to point out that this penalization does not prevent the extraction of overlapped rules because the examples covered by previously obtained fuzzy rules are not eliminated and take part in the computation of the confidence measure. In subgroup discovery algorithms, the possibility of extracting information on described examples is not eliminated since redundant descriptions of subgroups can show the properties of groups from a different perspective.

```
START
   Choose a target feature At_TAR
   Rule Set ← Ø
   REPEAT
    Execute the GA (At_TAR) obtaining rule R
    Local Search (R)
    If (confidence(R)≥ minimum confidence and R represents new examples)
     Rule Set ← Rule Set ∪ R
     Mark the set of examples covered by R
   WHILE (confidence(R) ≥ minimum confidence and
        R represents new examples)
END
```

Fig. 6. Iterative rule extraction model.

The scheme of the extraction model is shown in Fig. 6.

As can be seen, the confidence of the obtained rule in each iteration must be higher than a previously specified minimum value. In descriptive induction algorithms, one of the fundamental problems, and partially significant to the quality of the obtained results, is the specification of the minimum confidence required for the rules to be extracted. This value depends greatly on the problem to be solved, and its choice is a problem that is still not completely solved. In [49], a method based on fuzzy logic for the setting of the minimum confidence level is described.

C. Comparison Between SDIGA and Other Subgroup Discovery Algorithms

To verify the applicability of the proposal, we have compared its results with the results of other subgroup discovery algorithms.

The comparison has been made using as reference the work of Lavrac *et al.* with CN2-SD [28]. The CN2-SD algorithm, as mentioned in Section II, is an algorithm for the extraction of rules describing subgroups, obtained modifying the CN2 algorithm for the extraction of classification rules. A brief description of this algorithm can be found in Appendix A.

For the experimental evaluation and comparison of the approach proposed, the data sets *breast-w* and *diabetes*, both containing medical data and available in the UCI repository,[1] have been used.

These datasets have the following characteristics.

- *Breast-W:* this breast cancer domain was obtained from the University Medical Centre, Institute of Oncology, Ljubljana, Yugoslavia, thanks to Zwitter and Soklic, who provided the data. This dataset has 699 instances. Each instance has nine categorical variables and a class attribute with one of two possible classes: *benign* or *malignant*. There are 458 instances of class *benign* (65.5%) and 241 of class *malignant* (34.5%).
- *Diabetes (Pima Indians Diabetes Database):* this problem attempts to carry out the diagnosis of a binary variable to deduce if the patient shows signs of diabetes according to the World Health Organization's criterion. This dataset has 768 instances, with eight numerical variables. The class attribute has two values: *tested negative* for diabetes and *tested positive*. There are 500 instances of class *tested negative* (65%) and 268 of class *tested positive* (35%).

[1]www.ics.uci.edu/~mlearn/MLRepository.html.

The diabetes data set contains numerical variables and is used to show the results of the fuzzy rules extracted by the proposal in comparison with other subgroup discovery algorithm. In addition, our proposal can also manage categorical variables, and the breast-w data set is used to show the behavior of this proposal with this kind of problems.

The experiments have been carried out in the same way as in [28] to allow the comparison of subgroup discovery algorithms: tenfold cross-validation for the error estimation (dividing the data set in ten partitions and obtaining ten different combinations formed by 90% of data for training and 10% for test).

As we have mentioned previously, each run of the iterative process obtains a variable number of rules, all corresponding to the same value of the target feature. So, the process must be repeated for each one of the values of the target feature. Finally, as our proposal is a nondeterministic approach, we have carried out five runs on each training/test set partition. After obtaining the rules with the SDIGA algorithm, the measures of coverage (Cov), support (Sup_1), size, significance (Sig), and unusualness (WRAcc) were calculated with the expressions indicated in Section II in order to make the comparison. The parameters used in the experiments are:

- population size: 100;
- maximum number of evaluations of individuals in each GA run: 10 000;
- mutation probability: 0.01;
- number of linguistic labels for the numerical variables: 3;
- quality measure weights for the fitness function:
  - w_1: 0.4;
  - w_2: 0.3.

As mentioned in Section III-A2, the specification of the weights for the fitness function depends on the expert knowledge of the characteristics and/or complexity of the problem to be solved. In this paper, and without this expert knowledge, we use values for these weights specified only by considering a slight promotion of the extraction of general rules.

Tables I and II show the results obtained by the proposal and in the work of Lavrac *et al.* [28]. The results shown for the proposal are the averages of the values obtained in the test partitions for all the runs.

The tables include the results obtained with the SDIGA algorithm for four minimum confidence values (named "SDIGA CfMin 0.6" for the SDIGA algorithm with a minimum confidence value of 0.6, and so on), the results for the CN2 algorithm

TABLE I
COMPARISON OF SUBGROUP DISCOVERY ALGORITHMS FOR BREAST-W DATA SET

Algorithm	COV	(sd)	SUP	(sd)	Siz	(sd)	SIG	(sd)	WRACC	(sd)
SDIGA CfMin 0.6	0.398	0.07	0.983	0.03	5.4	0.88	16.910	3.81	0.113	0.03
SDIGA CfMin 0.7	0.414	0.07	0.981	0.02	5.2	0.74	17.399	4.05	0.116	0.03
SDIGA CfMin 0.8	0.435	0.09	0.969	0.03	4.5	1.36	18.523	5.81	0.124	0.03
SDIGA CfMin 0.9	0.478	0.07	0.923	0.07	2.4	0.81	24.434	6.63	0.156	0.03
CN2 WRAcc	0.150	0.04	0.900	0.02	8.8	0.95	13.300	1.69	0.063	0.04
CN2-SD (γ=0.5)	0.208	0.05	0.890	0.09	7.9	0.50	27.100	3.37	0.095	0.02
CN2-SD (γ=0.7)	0.174	0.04	0.840	0.04	8.5	1.75	2.100	0.02	0.079	0.01
CN2-SD (γ=0.9)	0.218	0.05	0.930	0.02	9.0	0.24	20.500	2.45	0.093	0.07
CN2-SD (add.)	0.260	0.04	0.860	0.05	9.2	1.24	26.600	3.43	0.111	0.04

TABLE II
COMPARISON OF SUBGROUP DISCOVERY ALGORITHMS FOR DIABETES DATA SET

Algorithm	COV	(sd)	SUP	(sd)	Siz	(sd)	SIG	(sd)	WRACC	(sd)
SDIGA CfMin 0.6	0.849	0.09	0.992	0.01	2.8	0.38	0.788	1.01	0.024	0.01
SDIGA CfMin 0.7	0.854	0.09	0.992	0.01	2.9	0.35	0.633	0.54	0.023	0.01
SDIGA CfMin 0.8	0.931	0.04	0.978	0.02	2.0	0.00	0.437	0.34	0.024	0.01
SDIGA CfMin 0.9	0.935	0.03	0.976	0.02	2.0	0.00	0.418	0.29	0.023	0.01
CN2 WRAcc	0.275	0.04	0.820	0.03	5.2	0.79	15.800	1.07	0.065	0.06
CN2-SD (γ=0.5)	0.296	0.06	0.920	0.06	6.0	0.68	14.900	1.95	0.085	0.07
CN2-SD (γ=0.7)	0.344	0.05	0.850	0.01	5.6	1.35	11.000	1.43	0.099	0.04
CN2-SD (γ=0.9)	0.299	0.05	0.950	0.01	5.4	0.30	15.200	1.85	0.086	0.07
CN2-SD (add.)	0.381	0.04	0.870	0.05	4.6	0.86	2.100	0.01	0.092	0.03

modifying the unusualness measure (CN2-WRAcc), and the results of the CN2-SD using different parameters for the weights (CN2-SD $(\gamma = x)$ is the CN2-SD algorithm using multiplicative weights with $\gamma = x$ and CN2-SD (add.) is the CN2-SD algorithm using additive weights). In Appendix A, the meaning of the CN2-SD parameters is described.

For each measure, the average value and the standard deviation (sd) are detailed. "COV" is the average coverage of the set of rules as measured in (11), "SUP" is the overall support of a set of rules as computed in (13), "Siz" is the number of rules in the induced set of rules, "SIG" is the average significance of a set of rules as measured in (15), and "WRACC" is the average rule unusualness as computed in (17).

The model performs better than the other algorithms for the measures coverage (COV), support (SUP), and size (Siz). This means that our proposal obtains a reduced set of rules with a high percentage of examples covered on average, a high number of examples satisfying both the antecedent and the consequent parts of the rules (i.e., a higher percentage of target positive examples leaving a smaller number of examples unclassified is covered), and with a low number of rules.

However, the results for interest measures show different behavior in the two problems: the significance (SIG) and unusualness (UNU) of our proposal are similar to the other algorithms for the first problem (breast-w) but are worse for the second one (diabetes).

Analyzing the results, we can observe that the use of different measures in the rule extraction process of CN2-SD with respect to SDIGA implies:

- the increase of the number of rules;
- the decrease of coverage and support; but
- the increase of the interest measurement values.

The inclusion of these measures (or adaptation of them to the fuzzy rules) can be considered in the improvement of SDIGA by means of a multiobjective version. This extension will be studied in a future extension of this proposal.

Another important decision is the number of labels per variable, because this can modify the rule behavior for support and interest measures. We have used three labels per rule to increase the linguistic interpretability of the model.

As main conclusions of this short comparison study, we can conclude that our proposal allows us to obtain subgroup discovering rules:

- with very high values of the measures of coverage and support, and so the rules can be considered very general and significantly representing the knowledge of the examples of the different values of the target variable;
- highly compact, because both the sizes of the set of rules and also the number of variables involved are small;
- highly descriptive, due to the use of fuzzy DNF rules, allowing a representation of the knowledge near to human reasoning and making the extracted knowledge very actionable, a main objective in any subgroup discovery algorithm;
- with a variable interest measure behavior.

IV. A CASE STUDY IN MARKETING: KNOWLEDGE DISCOVERY IN TRADE FAIRS WITH SDIGA

In the area of marketing, and specifically in the planning of trade fairs, it is important to extract conclusions from the information on previous trade fairs to determine the relationship between the trade fair planning variables and the success of the stand. For this problem a subgroup discovery rule induction algorithm is well suited. This problem of the extraction of useful information from trade fairs has been analyzed in the Department of Organization and Marketing, University of Mondragón, Spain [50].

Businesses consider trade fairs to be an instrument which facilitates the attainment of commercial objectives such as contact

TABLE III
CLASS DISTRIBUTION IN THE DATA SET

Class	# Instances	%
Low efficiency	38	16.5
Medium efficiency	148	65.0
High efficiency	42	18.5

with current clients, securing of new clients, taking orders, and improvement of the company image, among others [51]. One of the main disadvantages of this type of trade fair is the high investment they imply in terms of both time and money. This investment sometimes coincides with a lack of planning, which emphasizes the impression that trade fairs are no more than an "expense" a business must accept for various reasons such as tradition, client demands, and to avoid giving the impression that things are going badly, among other factors [52]. Therefore, the automatic extraction of information about the relevant variables that permit the attainment of unknown data, which partly determines the efficiency of the stands of a trade fair, is useful.

Anderson [53] proposes the use of the achievement of objectives set for the trade fair as an index to measure trade fair efficiency. Nevertheless, the percentage of exhibitors who have written objectives or who can express them in measurable terms is small. The absence of a formal document containing the goals of the companies makes it very difficult to quantify the degree of success of the fair. Therefore, it becomes necessary to use the valuation of the degree of attainment of a trade fair made by the company.

From a review of the literature and by asking the exhibitors, a questionnaire was designed to reflect the variables that allow a better explanation of trade fair success, later contrasted by experts. This questionnaire contains 104 variables, seven of which are numerical and the rest are categorical features (obtained by the experts by means of discretization). The questionnaire contains questions relating to the prior planning of the fair (which must be answered before the celebration of the fair) to the valuations on the participation in the fair as well as the actions to develop by the company after the fair (which will be answered once the fair has finished), and other questions to be answered during the fair.

In this way, once the data for each exhibitor have been gathered, the stand's global efficiency is rated as *high, medium* or *low,* in terms of the level of achievement of objectives set for the trade fair, based on various marketing criteria.

The data contained in this data set were collected in the Machinery and Tools biennial fair held in Bilbao, Spain, in March 2002 and contain information on 228 exhibitors. With the data collected for each exhibitor, the stands were characterized according to their level of achievement of objectives, obtaining the class distribution *(low, medium,* or *high efficiency)* shown in Table III.

For this real problem, the data mining algorithm should extract information of interest about each of the three efficiency groups of stands. The rules generated will determine the influence the different fair planning variables have over the results obtained by the exhibitor, therefore allowing fair-planning policies to be improved.

TABLE IV
RESULTS FOR LOW, MEDIUM, AND HIGH EFFICIENCY

Efficiency	# Var.	Support	Confidence
Low	5	0.029	1.000
	5	0.029	1.000
	4	0.114	1.000
	7	0.029	1.000
Medium	5	0.086	1.000
	5	0.023	1.000
	3	0.016	1.000
	2	0.008	1.000
	2	0.578	0.667
	3	0.047	1.000
High	3	0.054	1.000
	4	0.027	1.000
	2	0.027	1.000
	4	0.027	1.000
	2	0.081	1.000
	4	0.027	1.000

The use of a subgroup discovery algorithm for this problem is well suited because in a subgroup discovery task, the objective is not to generate a set of rules that cover all the data set examples but individual rules that, given a property of interest of the data, describe the most interesting subgroups for the user. This is the type of knowledge we want to obtain.

A. Results of the Experiments on the Marketing Data Set

Again, the experiments were carried out with five runs (five runs for each class of the target variable: low, medium, and high efficiency), and using the following parameters:

- population size: 100;
- maximum number of evaluations of individuals in each GA run: 10 000;
- mutation probability: 0.01;
- number of linguistic labels for the numerical variables: 3;
- quality measure weights for the fitness function:
  - $w_1 : 0.4$;
  - $w_2 : 0.3$;
- minimum confidence value: 0.6.

Table IV shows the best results obtained for all the classes of the target variable (low, medium, and high efficiency). In this table, the number of variables involved in each rule (# Var.) and the Support (Sup_2) and Confidence (Conf) of each rule are shown. The confidence is calculated as described in (18) and the support is computed as in (19).

The values of *Support* and *Confidence* are between zero and one. High values in support means that the rule covers most of the examples of the class, and high values in confidence means that the rule has few negative examples.

The rules generated have adequate values of confidence and support. The algorithm induces a set of rules with high confidence (higher than the minimum confidence value).

The rule support, except for some rules, is low. The market problem used in this paper is a difficult real problem in which inductive algorithms tend to obtain small disjuncts (specific rules that represent a small number of examples), more common in data sets than one might think at first glance. However, the small disjunct problem is not a determining factor in the induction process for subgroup discovery. This is because partial relations, i.e., subgroups with interesting characteristics, with a significant deviation from the rest of the data set, are sufficient.

TABLE V
RULES FOR LOW EFFICIENCY

# Rule	Rule
1	IF (Employees = (Huge OR High OR Normal OR Very Few) AND Annual sales = (Very Huge OR Huge OR High OR Few) AND Gratefulness pamphlet = Only to quality contacts AND Bar = No AND Food/Drink = Yes) THEN Efficiency = Low
2	IF (Kind of tracking of contacts = All AND Thank-you letter = No AND Stand with different heights = No AND Stewardesses = Yes AND Bar = No) THEN Efficiency = Low
3	IF (Zone = (North OR South) AND Important improvement image of the company = Medium AND Thank-you letter = No AND Stand with different heights = No) THEN Efficiency = Low
4	IF (Zone = (East OR South) AND Employees = (Very High OR High OR Normal OR Few) AND Annual sales = (Very High OR Normal OR Few) AND Thank-you letter = NO AND Contact tracking = (No OR All) AND Carpet = No AND Bar = No) THEN Efficiency = Low

TABLE VI
RULES FOR MEDIUM EFFICIENCY

# Rule	Rule
1	IF (Zone = (North OR Center OR South) AND Sector = (Starting OR Deformation OR Accessories OR CAD_CAM) AND Thank-you letter = All AND Thank-you pamphlet = (No OR Only Quality) AND Bar = Yes) THEN Efficiency = Medium
2	IF (Employees = (Huge OR Normal OR Very Few) AND Important quality contacts = (High OR Very High) AND Carpet = Yes AND Stewardesses = No AND Bar = No) THEN Efficiency = Medium
3	IF (Telephone calls = No AND Bar = Yes AND Food/Drink = Yes) THEN Efficiency = Medium
4	IF (Zone = Center AND Stewardesses = Yes) THEN Efficiency = Medium
5	IF (Important extracted information = (Very Low OR Low OR Medium OR High) AND Food/Drink = No) THEN Efficiency = Medium
6	IF (Zone = North AND Important improvement company image= (Medium OR High) AND Stewardesses = Yes) THEN Efficiency = Medium

TABLE VII
RULES FOR HIGH EFFICIENCY

# Rule	Rule
1	IF (Employees = (High OR Normal) AND Annual sales = (Very Huge OR Few) AND Thank-you pamphlet = (No OR Only quality) THEN Efficiency = High
2	IF (Thank-you letter = (No OR Only quality) AND Columns = Yes AND Bar = No AND Food/Drink = Yes THEN Efficiency = High
3	IF (Zone = Center AND Thank-you pamphlet = No) THEN Efficiency = High
4	IF (Employees= (Huge OR Very High OR High OR Very Few) AND Satisfaction public relations = (Very Low OR Medium OR Very High) AND Columns = Yes AND Food/Drink = No) THEN Efficiency = High
5	IF (Satisfaction improvement company image = (Low OR Very High) AND Telephone calls = No) THEN Efficiency = High
6	IF (Employees = Huge OR Normal) AND Publicity in exhibitor's catalogue = Yes AND Bar = Yes AND Food/Drink = No) THEN Efficiency = High

The knowledge discovered for each one of the target variable values is understandable by the user due to the use of DNF fuzzy logic and the low number of rules and conditions in the rule antecedents (below 10% of the 104 variables). Moreover, the rules obtained with the SDIGA algorithm are very simple, due to the application of a hill-climbing algorithm, which optimizes each extracted rule and increases their simplicity.

Tables V–VII show the rules corresponding to the quality values showed in Table IV. It must be noted that only seven of the features of the data set are numerical and the rest are categorical. We use three linguistic labels for the numerical variables (such as the *Size of the Stand*), but the categorical variables (such as *Employees*) have different numbers of possible values.

Marketing experts from the Department of Organization and Marketing, University of Mondragón, Spain, analyzed the results obtained and indicated the following.

- The exhibitors who obtained worse results were from the South zone, do not perform contact tracking, and, therefore, cannot optimize (closing a sale or giving more information after the show) the contacts they made at the trade show. Apart from that, the trade fair was held in the North zone, and the exhibitors were coming principally from that zone. So worse results for the exhibitors coming from the more distant zone can be explained due to the distance and the low level of knowledge of the peculiarities of the exhibition.

- The exhibitors who obtained best results *(high efficiency)* are those that are from the Central zone and do not send a thank-you pamphlet to all the contacts. These are large- or medium-sized companies, with either a very high or a low annual sales volume.
- Also, the exhibitors that obtained the best results have a very high or a small sales volume. The biggest companies can invest lots of money preparing the trade fair, so the results they get are better. Small companies spend little money on the trade show, and therefore their expectations about their participation are poorer. But, if they obtain good results, better than they expected, the evaluation of their performance at the trade show is very high.

V. Conclusion

This work presents a GFS for the extraction of subgroup discovery fuzzy rules. The proposal includes a GA in an iterative process to extract descriptive fuzzy rules, with different advantages.

- It obtains a reduced set of fuzzy rules.
- The extracted rules are interpretable due to the use of linguistic variables for numerical ones and due to the number of variables that play a role in each rule (see the marketing problem with the average of 4.3 variables per rule).
- Fuzzy logic allows the user to incorporate directly linguistic knowledge in the data mining process, to mix this knowledge with nonlinguistic information for categorical variables with nonnumerical values.
- The use of DNF rules gives a more flexible structure to the rules, allowing each variable to take more than one value. This type of rule structure allows us to describe the extracted knowledge more flexibly and, moreover, to make changes in the initial granularity in each rule in a descriptive way.
- The algorithm allows us to describe knowledge of different zones of the problem space of examples, due to the iterative mechanism with penalty (but no elimination) of the examples covered. This mechanism is similar to the incorporation of a weighting scheme in the examples to modify the covering algorithm when adapting classification rule algorithms for subgroup discovery. In spite of the penalty applied to extract different rules, the algorithm allows us to obtain overlapped rules describing knowledge from different perspectives because the examples covered are not eliminated.

The results of the proposal have been compared with the results of other subgroup discovery algorithms. The algorithm shows good behavior comparing its results with the same quality measures, in spite of the fact that our rule extraction method is not guided by the same quality measures used by the other algorithms. As we have mentioned, in future work we will analyze the extension of the algorithm using multiobjective genetic algorithms [54], [55], analyzing the meaning of Pareto-optimal solutions from the subgroup discovery point of view.

The proposal has been applied to the problem of knowledge extraction in trade fairs, and experts have established the validity of the extracted knowledge. This knowledge has allowed the experts to obtain novel conclusions on the available data.

Appendix A
CN2-SD Subgroup Discovery Algorithm

CN2-SD is a subgroup discovery algorithm obtained by adapting a standard classification rule-learning approach, CN2, to subgroup discovery. The proposed approach performs subgroup discovery through the following modifications of CN2:

a) replacing the accuracy-based search heuristic with a new weighted relative accuracy heuristic that trades off generality and accuracy of the rule;
b) incorporating example weights into the covering algorithm;
c) incorporating example weights into the weighted relative accuracy search heuristic;
d) using probabilistic classification based on the class distribution of covered examples by individual rules, both in the case of unordered sets of rules and ordered decision lists.

Next we will describe the main modifications of the CN2 algorithm, making it appropriate for subgroup discovery: the implementation of the weighted covering algorithm and the incorporation of example weights into the weighted relative accuracy heuristic.

1) Weighted Covering Algorithm: One of the problems of standard rule learners, such as CN2 and RIPPER, when used for subgroup discovery is the use of the covering algorithm for the construction of the set of rules because only the first few induced rules may be of interest as subgroup descriptions with sufficient coverage and significance. In the subsequent iterations of the covering algorithm, rules are induced from biased example subsets, i.e., subsets including only positive examples that are not covered by previously induced rules, which inappropriately bias the subgroup discovery process.

To avoid this problem, CN2-SD proposes the use of a weighted covering algorithm [27], in which the subsequently induced rules also represent interesting and sufficiently large subgroups of the population. The weighted covering algorithm modifies the classical covering algorithm in such a way that covered positive examples are not deleted from the current training set. Instead, in each run of the covering loop, the algorithm stores with each example a count indicating how many rules the example has been covered with so far. Weights derived from these example counts then appear in the computation of WRAcc. Initial weights of all positive examples E^s equal one, $\mathrm{w}(E^s, 0) = 1$, meaning that the example has not been covered by any rule. Each time a rule covers an example, the example weight will be decreased, so the uncovered target class examples whose weights have not been decreased will have a greater chance to be covered in the following iterations of the algorithm.

It is necessary to specify the weighting scheme, i.e., how the weight of each example decreases with the increasing number of covering rules. CN2-SD can use one of two weighting schemes.

- *Multiplicative weights,* where weights decrease multiplicatively. For a given parameter $0 < \gamma < 1$, a positive example E^s covered by i rules will have a weight γ^i. When $\gamma = 1$, the algorithm will always find the same rule over and over again, whereas with $\gamma = 0$, the algorithm would perform the same as the standard CN2 covering algorithm.

- *Additive weights,* where a positive example covered by i rules will have a weight $1/(i+1)$. In the first iteration, all target class examples have a weight of one, while in the following iterations the weight of each example is inversely proportional to its coverage by previously induced rules.

2) Modified WRAcc Heuristic with Example Weights: The CN2-SD proposal uses as a search heuristic a modified weighted relative accuracy to handle example weights. This provides the means to consider different parts of the example space in each iteration of the weighted covering algorithm.

In the WRAcc computation, all probabilities are computed by relative frequencies. An example weight measures how important it is to cover this example in the next iteration. The modified WRAcc measure is then defined as follows:

$$\mathrm{WRAcc}(\mathrm{Cond} \rightarrow \mathrm{Class}) = \frac{n'(\mathrm{Cond})}{N'} \cdot \left(\frac{n'(\mathrm{Class.Cond})}{n'(\mathrm{Cond})} - \frac{n'(\mathrm{Class})}{N'} \right). \quad (22)$$

In this equation, N' is the sum of the weights of all examples, $n'(\mathrm{Cond})$ is the sum of the weights of all covered examples, and $n'(\mathrm{Class.Cond})$ is the sum of the weights of all correctly covered examples.

To add a rule to the generated set of rules, the rule with the maximum WRAcc measure is chosen out of those rules in the search space, which are not yet present in the set of rules produced so far.

REFERENCES

[1] P. Clark and T. Niblett, "The cn2 induction algorithm," *Machine Learn.*, vol. 3, no. 4, pp. 261–283, Mar. 1989.

[2] D. Michie, D. J. Spiegelhalter, and C. C. Taylor, *Machine Learning, Neural and Statistical Classification*. London, U.K.: Ellis Horwood, 1994.

[3] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and I. Verkamo, "Fast discovery of association rules," in *Advances in Knowledge Discovery and Data Mining*, U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds. Menlo Park, CA: AAAI, 1996, pp. 307–328.

[4] W. Klösgen, "Explora: A multipattern and multistrategy discovery assistant," in *Advances in Knowledge Discovery and Data Mining*, U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds. Menlo Park, CA: AAAI, 1996, pp. 249–271.

[5] S. Wrobel, "An algorithm for multi-relational discovery of subgroups," in *Proc. 1st Eur. Symp. Principles Data Mining Knowl. Discovery (PKDD'97)*, Berlin, Germany, 1997, pp. 78–87.

[6] L. D. Raedt and L. Dehaspe, "Clausal discovery," *Machine Learn.*, vol. 26, pp. 99–146, 1997.

[7] P. A. Flach and I. Savnik, "Database dependency discovery: A machine learning approach," *AI Commun.*, vol. 12, pp. 139–160, 1999.

[8] D. Gamberger, N. Lavrac, and G. Krstacic, "Active subgroup mining: A case study in coronary heart disease risk group detection," *Artif. Intell. Med.*, vol. 28, pp. 27–57, 2003.

[9] D. Gamberger, N. Lavrac, F. Zelezny, and J. Tolar, "Induction of comprehensible models for gene expression datasets by subgroup discovery methodology," *J. Biomed. Inform.*, vol. 37, pp. 269–284, 2004.

[10] N. Lavrac, B. Cestnik, D. Gamberger, and P. Flach, "Decision support through subgroup discovery: Three case studies and the lessons learned," *Machine Learn.*, vol. 57, pp. 115–143, 2004.

[11] B. Kavsek and N. Lavrac, "Using subgroup discovery to analyze the UK traffic data," *Metodoloski zvezki*, vol. 1, pp. 249–264, 2004.

[12] D. Dubois, H. Prade, and T. Sudkamp, "On the representation, measurement, and discovery of fuzzy associations," *IEEE Trans. Fuzzy Syst.*, vol. 13, pp. 250–262, 2005.

[13] H. Ishibuchi, T. Nakashima, and M. Nii, *Classification and Modeling with Linguistic Information Granules*. Berlin, Germany: Springer-Verlag, 2004.

[14] G. Chen and Q. Wei, "Fuzzy association rules and the extended mining algorithms," *Inform. Sci.*, vol. 147, pp. 201–228, 2002.

[15] T. P. Hong, K. Y. Liu, and S. L. Wang, "Fuzzy data mining for interesting generalized association rules," *Fuzzy Sets Syst.*, vol. 138, pp. 255–269, 2003.

[16] E. Hüllermeier, "Fuzzy methods in machine learning and data mining: Status and prospects," *Fuzzy Sets Syst.*, vol. 156, pp. 387–407, 2005.

[17] C. H. Wang, J. F. Liu, T. P. Hong, and S. S. Tseng, "A fuzzy inductive learning strategy for modular rules," *Fuzzy Sets Syst.*, vol. 103, pp. 91–105, 1999.

[18] Y. Yuang and M. J. Shaw, "Induction of fuzzy decision trees," *Fuzzy Sets Syst.*, vol. 69, pp. 125–139, 1995.

[19] D. Nauck, F. Klawonn, and R. Kruse, *Foundations of Neuro-Fuzzy Systems*. New York: Wiley, 1997.

[20] O. Cordón, F. Herrera, F. Hoffmann, and L. Magdalena, *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. Singapore: World Scientific, 2001.

[21] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[22] A. A. Freitas, *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Berlin, Germany: Springer-Verlag, 2002.

[23] M. L. Wong and K. S. Leung, *Data Mining Using Grammar Based Genetic Programming and Applications*. Norwell, MA: Kluwer Academics, 2000.

[24] W. Klösgen, "Subgroup discovery," in *Handbook of Data Mining and Knowledge Discovery*, W. Klösgen and J. Zytkow, Eds. New York: Oxford Univ. Press, 2002, pp. 354–364.

[25] G. Piatetsky-Shapiro and C. Matheus, "The interestingness of deviation," in *Proc. AAAI Workshop Knowl. Discovery Databases*, Seattle, 1994, pp. 25–35.

[26] A. Silberschatz and A. Tuzhilin, "On subjective measures of interestingness in knowledge discovery," in *Proc. 1st Int. Conf. Knowl. Discovery Data Mining (KDD-95)*, Menlo Park, CA, 1995, pp. 275–281.

[27] D. Gamberger and N. Lavrac, "Expert-guided subgroup discovery: Methodology and application," *J. Artif. Intell. Res.*, vol. 17, pp. 1–27, 2002.

[28] N. Lavrac, B. Kavsec, P. Flach, and L. Todorovski, "Subgroup discovery with CN2-SD," *J. Machine Learning Res.*, vol. 5, pp. 153–188, 2004.

[29] N. Lavrac, P. Flach, and B. Zupan, "Rule evaluation measures: A unifying view," in *Proc. 9th Int. Workshop Inductive Logic Program. (ILP'99)*, Bled, Slovenia, 1999, pp. 174–185.

[30] J. R. Quinlan, "Generating production rules from decision trees," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, Milan, Italy, 1987, pp. 304–307.

[31] O. Cordón, M. J. del Jesus, and F. Herrera, "Genetic learning of fuzzy rule-based classification systems co-operating with fuzzy reasoning methods," *Int. J. Intell. Syst.*, vol. 13, pp. 1025–1053, 1998.

[32] W. M. Klösgen and M. May, "Census data mining—An application," in *Proc. ECML/PKDD'02 Workshop Mining Official Data (MOD'02)*, Helsinki, Finland, 2002, pp. 65–79.

[33] N. Lavrac, F. Zelezny, and P. A. Flach, "RSD: Relational subgroup discovery through first-order feature construction," in *Proc. 12th Int. Conf. Inductive Logic Program. (ILP 2002)*, Sydney, Australia, 2002, pp. 149–165.

[34] B. Kavsek, N. Lavrac, and V. Jovanoski, "APRIORI-SD: Adapting association rule learning to subgroup discovery," in *Proc. 5th Int. Symp. Intell. Data Anal. (IDA 2003)*, Berlin, Germany, 2003, pp. 230–241.

[35] R. Agrawal, T. Imielinski, and A. N. Swami, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Washington, DC, 1993, pp. 207–216.

[36] V. Jovanoski and N. Lavrac, "Classification rule learning with APRIORI-C," in *Proc. 10th Portuguese Conf. Artif. Intell. (EPIA 2001)*, Porto, Portugal, 2001, pp. 44–51.

[37] M. Atzmueller, F. Puppe, and H.-P. Buscher, "Towards knowledge-intensive subgroup discovery," in *Proc. Workshop Lernen Wissensentdeckung Adaptivitat (LWA 2004)*, Berlin, Germany, 2004, pp. 111–117.

[38] T. Bäck, D. Fogel, and Z. Michalewicz, *Handbook of Evolutionary Computation*. Oxford, U.K.: Oxford Univ. Press, 1997.

[39] O. Cordón, F. A. C. Gomide, F. Herrera, F. Hoffmann, and L. Magdalena, "Ten years of genetic fuzzy systems: Current framework and new trends," *Fuzzy Sets Syst.*, vol. 14, pp. 5–31, 2004.

[40] B. Carse, T. C. Fogarty, and A. Munro, "Evolving fuzzy rule based controllers using genetic algorithms," *Fuzzy Sets Syst.*, vol. 80, pp. 273–293, 1996.

[41] C.-H. Wang, T.-P. Hong, and S.-S. Tseng, "Integrating fuzzy knowledge by genetic algorithms," *IEEE Trans. Evol. Comput.*, vol. 2, pp. 138–149, 1998.

[42] T. Kovacs, *Strength or Accuracy: Credit Assignment in Learning Classifier Systems*. Berlin, Germany: Springer-Verlag, 2004.
[43] H. Ishibuchi, T. Nakashima, and T. Murata, "Performance evaluation of fuzzy classifier systems for multidimensional patern classification problems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, pp. 601–618, 1999.
[44] A. Parodi and P. Bonelli, "A new approach to fuzzy classifier systems," in *Proc. 5th Int. Conf. Genetic Algorithms (ICGA'93)*, San Mateo, 1993, pp. 223–230.
[45] T.-P. Hong and Y.-C. Lee, "Mining coverage-based fuzzy rules by evolutional computation," in *Proc. 2001 IEEE Int. Conf. Data Mining (ICDM 2001)*, San Jose, CA, 2001, pp. 218–224.
[46] A. González and R. Pérez, "SLAVE: A genetic learning system based on an iterative approach," *IEEE Trans. Fuzzy Syst.*, vol. 7, pp. 176–191, 1999.
[47] O. Cordón, M. J. del Jesus, F. Herrera, and M. Lozano, "MOGUL: A methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning approach," *Int. J. Intell. Syst.*, vol. 14, pp. 1123–1153, 1999.
[48] A. Giordana and F. Neri, "Search-intensive concept induction," *Evol. Comput.*, vol. 3, pp. 375–416, 1995.
[49] S. Zhang, J. Lu, and C. Zhang, "A fuzzy logic based method to acquire user threshold of minimum-support for mining association rules," *Inform. Sci.*, vol. 164, pp. 1–16, 2004.
[50] M. Mesonero, "Towards an effective model of trade fair planning based on genetic algorithms," (in Spanish) Ph.D. dissertation, Dept. Organization and Marketing, Mondragón Univ., Mondragón, Spain, 2004.
[51] S. Gopalakrishna, G. L. Lilien, J. D. Williams, and I. K. Sequeira, "Do trade shows pay off," *J. Marketing*, vol. 59, pp. 75–83, 1995.
[52] S. Millar, *How to Get the Most of the Trade Shows*. Lincolnwood, IL: NTC Publishing, 2003.
[53] S. Anderson, "Better measurement: Help exhibitors identify their return on objectives," *EXPO Mag.*, p. 21, 2002.
[54] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester, U.K.: Wiley, 2001.
[55] C. A. Coello, D. A. Van Veldhuizen, and G. B. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*. New York: Kluwer Academic, 2002.

María José del Jesus received the M.Sc. and Ph.D. degrees in computer science from the University of Granada, Spain, in 1994 and 1999, respectively.

She is an Associate Professor with the Department of Computer Science, University of Jaén, Spain. Her research interests include fuzzy rule-based systems, fuzzy and linguistic modeling, fuzzy classification, machine learning, genetic fuzzy systems, and data mining.

Pedro González received the M.Sc. degree in computer science from the University of Granada, Spain, in 1991.

Currently, he is an Associate Professor with the Department of Computer Science, University of Jaén, Spain. His research interests include fuzzy rule-based systems, machine learning, genetic fuzzy systems, subgroup discovery, and data mining.

Francisco Herrera received the M.Sc. and Ph.D. degrees in mathematics from the University of Granada, Spain, in 1988 and 1991, respectively.

He is currently a Professor in the Department of Computer Science and Artificial Intelligence, University of Granada. He has published more than 100 papers in international journals and is coauthor of *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases* (Singapore: World Scientific, 2001). He has coedited three international books and 15 special issues in international journals on different soft computing topics, such as, preference modelling, computing with words, genetic algorithms, and genetic fuzzy systems. He currently serves on the editorial boards of *Soft Computing, Mathware and Soft Computing, International Journal of Hybrid Intelligent Systems, International Journal of Computational Intelligence Research, International Journal of Information Technology and Intelligent Computing,* and *Evolutionary Computation.* His current research interests include computing with words, preference modelling and decision making, data mining and knowledge discovery, genetic algorithms, and genetic fuzzy systems.

Mikel Mesonero received the Ph.D. degree in business administration from the University of Mondragón, Spain, in 2004.

He is currently an Associate Professor with the Department of Marketing, University of Mondragón, where he currently is Head of the department. He has coordinated several research projects in marketing subjects. His current main research interests are in the fields of customer relationship management and branding.