An analysis of evolutionary algorithms with different types of fuzzy rules in subgroup discovery

Cristóbal José Carmona, Pedro González, María José del Jesus, and Francisco Herrera

***Abstract*—The interpretability of the results obtained and the quality measures used both to extract and evaluate the rules are two key aspects of Subgroup Discovery. In this study, we analyse the influence of the type of rule used to extract knowledge in Subgroup Discovery, and the quality measures more adapted to the evolutionary algorithms for Subgroup Discovery developed so far. The adaptation of the NMEF-SD algorithm to extract disjunctive formal norm rules is also presented.**

I. Introduction

Data mining (DM) is the stage within Knowledge Discovery in Databases (KDD) [1] responsible for high level automatic knowledge discovery using real data. Two approaches can be distinguished in the DM process: predictive induction, whose objective is the discovery of knowledge for classification or prediction [2], and descriptive induction, whose main objective is the extraction of interesting knowledge from the data. This work focuses on subgroup discovery (SD) [3], a descriptive DM task including some features of predictive DM. It can be considered that SD is between the extraction of association rules and the obtaining of classification rules. The goal of SD is the discovery of interesting individual patterns in relation to a specific property which is of interest to the user.

The interpretability of the results obtained is an important issue in SD because the goal of the SD task is to find significant, relevant and previously unknown information about groups of interest. In this sense, rules are a suitable tool for the representation of knowledge in the extraction of information describing subgroups. This is the reason we are interested in the description of subgroups through rules.

The use of genetic algorithms (GAs) [4] and fuzzy logic [5] is interesting for the SD task. GAs explore the search space thoroughly and handle the relations between variables appropriately, and therefore develop searches particularly suited to rule extraction. Fuzzy logic, and particularly the use of descriptive fuzzy rules, allows us to represent and use knowledge in a similar way to human reasoning. With the use of fuzzy rules we obtain more interpretable and actionable solutions in the field of SD, and in general in the analysis of data in order to establish relationships and identify patterns [6]. So we are interested in the extraction of fuzzy rules to describe subgroups and in the use of GAs to obtain this type of rule.

C.J. Carmona, P. González, and M.J. del Jesus are with the Department of Computer Science, University of Jaen, 23071 Jaen, Spain (e-mail: {ccarmona|mjjesus|pglez}@ujaen.es).

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

In the design of any algorithm for the extraction of rules for SD there are different questions to consider: the type of rule used to represent the knowledge, the quality measures used to evaluate the rules and the way to consider them in the DM process.

There are different types of fuzzy rules: canonical rules, disjunctive normal form (DNF) rules, rules with degrees of certainty or rules with weights [7]. For SD, fuzzy rules with weights or degrees of certainty are not considered because they are less interpretable.

The quality measures used to obtain and evaluate rule sets is a very important issue in SD. There are different measures which can be used for this purpose, but currently there is no consensus on which of them are most appropriate for this rule induction task.

The extraction of fuzzy rules for SD can be considered as a multi-objective problem rather than a single objective one, since –as mentioned– there are different quality measures which can be used for evaluating a rule in SD. In the literature there are several evolutionary proposals for the SD task; one of these, SDIGA [8], uses an aggregation of the objective functions, and another, MESDIF [9], uses elitism in the multi-objective evolutionary search.

The main objective of our work is to analyse the influence of the type of rule used to represent knowledge and the quality measures used to evaluate rules, in the context of the Genetic Fuzzy System (GFS) for SD developed up to the moment. To complete this objective, we also present the extension of the NMEF-SD algorithm [10] in order to use not only canonical but also DNF rules.

The paper is organised as follows: In Section II, SD and GFSs in SD are presented. The evolutionary approach for SD using canonical and DNF rules is explained in Section III. In Section IV the results obtained with the evolutionary algorithms are analysed. Finally, conclusions are outlined in Section V.

II. Preliminaries

In this section the SD task and some considerations on the use of GFSs in rule induction processes focusing on the type of approach best adapted for SD are briefly described.

A. Subgroup Discovery

The concept of SD was initially formulated by Klösgen [11] and Wrobel [12]: Given a population of individuals and a property of those individuals we are interested in, find population subgroups that are statistically "most interesting", e.g., are as large as possible and have the most unusual

978-1-4244-3597-5/09/$25.00 ©2009 IEEE

statistical characteristics with respect to the property of interest.

Therefore, the objective in SD is to discover characteristics of the subgroups by constructing simple rules with high support and significance.

A rule R_i can be described as:

$$R_i : Cond_i \rightarrow Class_j$$

where the antecedent describes the subgroup in canonical or DNF form.

To describe a fuzzy rule, we consider a SD problem with:

- $\{X_m/m = 1, \ldots, n_v\}$, a set of features used to describe the subgroups, where n_v is the number of features. These variables can be categorical or numerical.
- $\{Class_j/j = 1, \ldots, n_c\}$, a set of values for the target variable, where n_c is the number of values.
- $\{E^k = (e^k_1, e^k_2, \ldots, e^k_{n_v})/k = 1, \ldots, N\}$, a set of examples, where $class_j$ is the value of the target variable for the example E^k (i.e., the class for this example) and N is the number of examples for the descriptive induction process.
- $X_m : \{LL^1_m, LL^2_m, \ldots, LL^{l_m}_m\}$, 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: the variable X_m has l_m different linguistic labels to describe its domain in an understandable way.

Then, a fuzzy rule in DNF form can be expressed as:

$$R_1 : \ If\ X_1 = (LL^1_1\ or\ LL^2_1)\ and\ X_6 = LL^3_6\ then\ Class_j$$

where LL^1_1 is the linguistic label number 1 of the variable number 1.

One of the most important aspects in SD is the quality measures used both to extract and evaluate the rules. As previously mentioned, there is no a consensus in the field about what are most adapted measures for the SD process, but the most used measures are:

- Significance [11]. Indicates the significance of a finding, if measured by the likelihood ratio of a rule:

$$Sig(Cond_i \rightarrow Class_j) = \qquad (1)$$

$$2 \cdot \sum_{k=1}^{n_c} n(Class_k \cdot Cond_i) \cdot log \frac{n(Class_k \cdot Cond_i)}{n(Class_k) \cdot p(Cond_i)}$$

  where $n(Class_k \cdot Cond_i)$ is the number of examples which satisfy the conditions for the antecedent and belong to $Class_k$, $n(Class_k)$ is the number of examples for the target variable indicated in the consequent part of the rule and $p(Cond_i)$ is used as a normalising factor.

- Unusualness [13]. Measures the balance between the coverage of the rule and its accuracy gain:

$$WRAcc(Cond_i \rightarrow Class_j) = \qquad (2)$$

$$= \frac{n(Cond_i)}{N} \left(\frac{n(Class_j \cdot Cond_i)}{n(Cond_i)} - \frac{n(Class_j)}{N} \right)$$

  where $n(Cond_i)$ is the number of examples which satisfy the antecedent and N is the number of examples. The weighted relative accuracy of a rule can be described as the coverage using the first part of the expression $(n(Cond_i)/N)$ and the accuracy gain using the second part $(n(Class_j \cdot Cond_i)/n(Cond_i)) - (n(Class_j)/N)$.

- Support [13]. Is defined as the frequency of correctly classified examples covered by the rule:

$$Sup_c N(R_i) \ = \ \frac{n(Class_j \cdot Cond_i)}{n(Class_j)} \qquad (3)$$

- Fuzzy Confidence [8]. Determines the relative frequency of examples which verify the complete rule among those which satisfy only the antecedent part:

$$FCnf(R_i) \ = \ \frac{\sum_{E^k \in E/E^k \in Class_j} APC(E^k, R_i)}{\sum_{E^k \in E} APC(E^k, R_i)} \qquad (4)$$

  where the antecedent part compatibility (APC) is the degree of compatibility between an example and the antecedent part of a fuzzy rule:

$$APC(E^k, R_i) = T(TC(\mu_{LL^1_1}(e^k_1), \ldots, \mu_{LL^{l_1}_1}(e^k_1)), \ldots,$$

$$TC(\mu_{LL^1_{n_v}}(e^k_{n_v}), \ldots, \mu_{LL^{l_{n_v}}_{n_v}}(e^k_{n_v}))) > 0 \qquad (5)$$

  where T is the $t-norm$ selected to represent the meaning of the AND operator (the fuzzy intersection, in our case the minimum) and TC is the $t-conorm$ selected to represent the meaning of the OR operator (the fuzzy union, in our case the maximum).

For a set of rules the value of each quality measure is computed as the average of the values for each rule.

B. Genetic fuzzy systems for subgroup discovery

A GFS is essentially a fuzzy system enhanced by a learning process based on a GA [14], [15]. Currently, GFSs are being applied to a wide range of real-world problems. The research related to this area is growing, and a number of open problems and future directions can be found in [16], [17], [18].

The genetic representation of solutions is the most determinant aspect of any GFS proposal. In this sense, the proposals in the specialised literature follow two approaches in order to encode rules within a population of individuals [14]: The “Chromosome = Rule” approach, in which each individual codifies a single rule; and the “Chromosome = Set of rules” approach, also called the Pittsburgh approach, in which each individual represents a set of rules.

There is a large body of literature which focuses on the extraction of fuzzy rules in descriptive data mining. This is widely applied to association rule extraction. The use of fuzzy sets in fuzzy rules 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 attribute domains. Proposals for the extraction of fuzzy association rules include [19], [20], [21], [22].

There are different evolutionary proposals in literature for extracting fuzzy rules in SD. This task can be considered as a multi-objective problem and the evolutionary proposals are represented with aggregation of the objective functions or with a multi-objective approach. The GFSs developed for the SD task are introduced below:

- *SDIGA* [8], [23] is an evolutionary fuzzy rule induction approach for SD which uses support and confidence as quality measures. A later version of this algorithm, *SDIGA-II*, instead uses support and unusualness as quality measures. This algorithm employs canonical and DNF representation.
- *MESDIF* [9] is a multi-objective evolutionary algorithm for SD based on the SPEA2 approach [24]. It considers linguistic fuzzy rules and defines support and confidence as quality measures. This algorithm uses canonical and DNF representation.
- *NMEF-SD* [10] is a multi-objective evolutionary algorithm which follows the NSGA-II [25] approach. This algorithm uses unusualness and support as quality measures and is implemented for obtaining canonical rules.

Next section describes the adaptation of the NMEF-SD algorithm for the obtaining of DNF rules. In this way, we can complete the study over the evolutionary algorithms for SD presented at the moment, with canonical and DNF representation of the rules.

III. NMEF-SD: Non-dominated Multi-objective Evolutionary algorithm based on the extraction of Fuzzy rules for Subgroup Discovery

NMEF-SD algorithm extracts descriptive fuzzy or crisp rules –depending on the nature of the features of the problem (continuous and/or nominal variables)– which describe subgroups.

When the features are continuous the algorithm uses fuzzy rules, and the fuzzy sets corresponding to the linguistic labels are defined by means of the corresponding membership functions. These can be specified by the user or defined by means of a uniform partition if expert knowledge is not available. In our work, uniform partitions with triangular membership functions are used as shown in Fig. 1 for a variable with five linguistic labels.

The objective of this evolutionary process is to extract a variable number of different rules which give information about the examples from the original set for each value of the target variable. As the objective is to obtain a set of rules which describe subgroups for all the values of the target variable, the algorithm must be executed as many times as the number of different values the target variable contains.

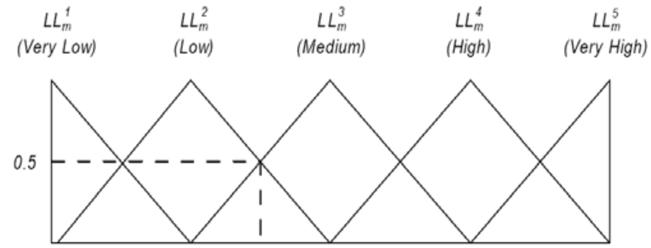

Fig. 1. Example of fuzzy partition for a continuous variable

Each candidate solution is codified according to the "chromosome = rule" approach, in which each individual codifies a single rule.

The extension presented in this study allows NMEF-SD to use not only canonical rules but also DNF rules. For a canonical rule the antecedent of a rule is composed of a conjunction of value-variable pairs, and the value 0 is used to indicate that the variable is not considered for the rule (Fig. 2). For a DNF rule, a fixed-length binary representation is used in which one bit for each of the possible values of every 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 value is used in the rule (Fig. 3).

$$\begin{array}{c} Genotype \\ \mid 2 \mid 0 \mid 1 \mid 0 \mid \\ \\ \Downarrow \\ Phenotype \end{array}$$

IF $(x_1 = Medium)$ AND $(x_3 = Low)$ THEN $(x_{Obj} = FixedValue)$

Fig. 2. Representation of a canonical rule in NMEF-SD

$$\begin{array}{c} Genotype \\ \left\| \begin{array}{c} \text{x1} \\ 1 \mid 1 \mid 0 \end{array} \right\| \begin{array}{c} \text{x2} \\ 0 \mid 0 \mid 0 \end{array} \left\| \begin{array}{c} \text{x3} \\ 1 \mid 0 \mid 0 \end{array} \right\| \\ \\ \Downarrow \\ Phenotype \end{array}$$

IF $(x_1 = (Low\ OR\ Medium))$ AND $(x_3 = Low)$ THEN $(x_{Obj} = FixedValue)$

Fig. 3. Representation of a DNF rule in NMEF-SD

In this extraction process the objective is to obtain interpretable rules with high precision and generality. To do so, unusualness (Eq. 2) and support (Eq. 3) are the quality measures considered in the algorithm.

NMEF-SD is based on the NSGA-II approach [25], and its main purpose is to evolve the population based on the non-dominated sort of the solutions in fronts of dominance. The first front is composed of the non-dominated solutions of the population (the Pareto front), the second is composed of the solutions dominated by one solution, the third of solutions dominated by two, and so on.

Fig. 4 shows the evolutionary algorithm of the NMEF-SD algorithm.

```
BEGIN
    Create P_0 with biased initialisation
    REPEAT
        Q_t ← ∅
        Tournament Selection (P_t)
        Q_tc ← Multi-point Crossover (P_t)
        Q_tm ← Biased Mutation (Q_tc)
        Q_t ← Q_tc + Q_tm
        Q_t ← Q_t+ offspring
        R_t ← Join(P_t,Q_t)
        Fast-non-dominated-sort(R_t)
        IF Pareto_front evolves
            Introduce fronts in P_{t+1}
        ELSE
            Re-initialisation based on coverage P_{t+1}
    WHILE (num-eval < Max-eval)
    RETURN Pareto_front
END
```

Fig. 4. The NMEF-SD algorithm

NMEF-SD tries to obtain a set rules with high precision, high generality and proper differentiation among them with different operators. Generality is obtained both with an operator which performs a biased initialisation process and with biased genetic operators, while diversity is introduced with the crowding distance [25] and with re-initialisation based on coverage.

The most important parts of the algorithm are next described:

- *Initialisation:* First step of the algorithm, which generates a biased population with a maximum of 75% of the total of the individuals generated with 25% of the variables in the rule. The remaining individuals (25%) are randomly generated.
- *Genetic operators:* Generate the offspring population. These operators are tournament selection [26], multi-point crossover [27] and biased mutation [8].
- *Fast-non-dominated sort*: Performs a sort in fronts of population based on non-dominance. The first front (F_1) is the Pareto front.
- *Re-initialisation based on coverage:* Performs a re-initialisation of the population, except the Pareto front, with individuals which cover new examples of the data set not previously covered. This operator is applied when the Pareto front does not evolve during a percentage (5%) of the maximum number of evaluations.
- *Stop condition:* Is determined for a maximum number of evaluations. At this point, the algorithm returns the set of rules which overcome a given confidence threshold.

IV. Experimentation

This study examines the evolutionary algorithms of SD described in the specialised bibliography in order to analyse the influence on the results of the type of rule, the quality measures used to evaluate the rules and the way these quality measures are considered in the evolutionary process. To do this, different data sets from the UCI repository [28] have been used. These data sets are classified in groups according to the type of variables: discrete with two classes, continuous with two classes, discrete with more than two classes, and continuous with more than two classes.

As the evolutionary algorithms are non-deterministic, they are run five times and a ten-fold cross validation is performed.

The parameters used in NMEF-SD are: population size of 25 individuals, maximum number of evaluations of 10000, crossover probability of 0.6 and mutation probability of 0.1. In MESDIF, SDIGA and SDIGA-II the population size is 100, crossover probability is 0.6, and the mutation probability is 0.01.

Tables I-IV show the average values obtained for the different data sets: number of rules ($\sharp Rul$), number of variables ($\sharp Var$), significance ($SIGN$, Eq. 1), unusualness ($WRAcc$, Eq. 2) , support (SUP_cN, Eq. 3) and fuzzy confidence ($FCNF$, Eq. 4). The best results are marked in bold characters. We consider that an algorithm stands out when its global results are the best in the quality measures described in Section II-A. The best algorithm is marked in the table with bold-italic characters in the name.

In our study an analysis of tables I-IV is performed with respect to:

1) *Type of rule.* The choice of the type of rule depends on the way the expert wishes to represent the knowledge. In the absence of preferences we can see:

- For data sets with discrete features either canonical and DNF representation can be used.
- For data sets with continuous features, it must be considered whether the algorithm is multi-objective or not. Canonical representation shows better results for multi-objective algorithms, and for mono-objective algorithms DNF representation obtains better results.

2) *Quality measures to be considered in the evolutionary process.* According to Eqs. 2-4:

- Significance ($SIGN$) is a statistical criterion which measures the significance of the antecedent part of the rule. It must be noted that it computes the distributional unusualness without bias toward any particular class, although the rule has specific class in the consequent.
- Unusualness ($WRAcc$) considers not only the distributional unusualness (as does Significance) but also the coverage of the rule.
- Support (SUP_cN) measures the percentage of examples belonging to the class indicated in the consequent which is described by the rule. It must be noted that this is a coverage measure with an accuracy component due to the fact that it considers only positive examples.
- Fuzzy confidence ($FCNF$) measures the rule accuracy.

For SD, a DM task between prediction and description, unusualness and support measures are the best choice because they represent a good balance between precision, interest and coverage. In tables I-IV it can be observed that algorithms NMEF-SD and SDIGA-II (which use these quality measures) obtain the best results.

3) *Evolutionary process.* As expected in any problem with different objectives, the evolutionary algorithms with a multi-objective approach obtain better results than those which consider an aggregation of the objectives.
In addition, considering only the multi-objective approaches, NMEF-SD is the algorithm which obtains the best results. This could be the result of the quality measures used and the structure of the evolutionary algorithm performed by NMEF-SD.
It must be highlighted that NMEF-SD obtains smaller rule sets, and so more interpretable ones, a very important characteristic for SD.

V. CONCLUSIONS

In this study, an analysis of the influence of the type of rule and the quality measures used in the context of GFSs for SD is developed. Moreover, the adaptation of the NMEF-SD algorithm to extract DNF rules is also presented.

An analysis of the results obtained with different data sets and different evolutionary algorithms for SD shows that unusualness and support are the most suitable quality measures to be included in a SD algorithm.

Furthermore, some tendencies in the type of fuzzy rules related with mono-objective and multi-objective approaches should be highlighted. The best results are obtained using multi-objective approaches with canonical representation. For mono-objective approaches the subgroup descriptions with DNF rules are the best choice.

The experiments show that the best results for all the data sets are obtained with the NMEF-SD algorithm.

ACKNOWLEDGMENT

This work was supported by the Spanish Ministry of Education, Social Policy and Sports under projects TIN-2008-06681-C06-01 and TIN-2008-06681-C06-02, and by the Andalusian Research Plan under project TIC-3928.

REFERENCES

[1] U. M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery: an overview," in *Advances in knowledge discovery and data mining*, 1996, pp. 1–34.
[2] D. Michie, D. J. Spiegelhalter, and C. C. Tayloy, *Machine Learning*. Ellis Horwood, 1994.
[3] D. Gamberger and N. Lavraĉ, "Expert-Guided Subgroup Discovery: Methodology and Application," *Journal Artificial Intelligence Research*, vol. 17, pp. 501–527, 2002.
[4] D. E. Golberg, "Genetic Algorithms in search, optimization and machine learning," *Addison-Wesley*, 1989.
[5] L. A. Zadeh, "The concept of a linguistic variable and its applications to approximate reasoning, Parts I, II, III," *Information Science*, vol. 8-9, pp. 199–249,301–357,43–80, 1975.
[6] E. Hüllermeier, "Fuzzy methods in machine learning and data mining: Status and prospects," *Fuzzy Sets and Systems*, vol. 156, no. 3, pp. 387–406, 2005.
[7] H. Ishibuchi, T. Nakashima, and M. Nii, *Classification and Modeling with Linguistic Information Granules: Advanced Approaches to Linguistic Data Mining*, ser. Advanced Information Processing. Springer, Berlin, 2005.
[8] M. J. del Jesus, P. González, F. Herrera, and M. Mesonero, "Evolutionary Fuzzy Rule Induction Process for Subgroup Discovery: A case study in marketing," *IEEE Transactions on Fuzzy Systems*, vol. 15, no. 4, pp. 578–592, 2007.
[9] F. J. Berlanga, M. J. del Jesus, P. González, F. Herrera, and M. Mesonero, "Multiobjective Evolutionary Induction of Subgroup Discovery Fuzzy Rules: A Case Study in Marketing," ser. LNCS, vol. 4065. Springer, 2006, pp. 337–349.
[10] C. J. Carmona, P. González, M. J. del Jesus, and F. Herrera, "Non-dominated Multi-objective Evolutionary algorithm based on Fuzzy rules extraction for Subgroup Discovery," in *HAIS 2009*, ser. LNAI, vol. 5572, pp. 573–580.
[11] W. Klösgen, "Explora: A Multipattern and Multistrategy Discovery Assistant," in *Advances in Knowledge Discovery and Data Mining*. Fayyad, U., et. al. Editors, 1996, pp. 249–271.
[12] S. Wröbel, "An Algorithm for Multi-relational Discovery of Subgroups," in *PKDD '97: Proceedings of the First European Symposium on Principles of Data Mining and Knowledge Discovery*, ser. LNCS, vol. 1263. Springer, 1997, pp. 78–87.
[13] N. Lavraĉ, B. Kavŝek, P. A. Flach, and L. Todorovski, "Subgroup Discovery with CN2-SD," *Journal of Machine Learning Research*, vol. 5, pp. 153–188, 2004.
[14] O. Cordón, F. Herrera, F. Hoffmann, and L. Magdalena, *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. World Scientific, 2001.
[15] F. Herrera, "Genetic fuzzy systems: taxomony, current research trends and prospects," *Evolutionary Intelligence*, vol. 1, pp. 27–46, 2008.
[16] J. Casillas and B. Carse, "Special issue on Genetic Fuzzy Systems: Recent Developments and Future Directions," *Soft Computing*, vol. 13, no. 5, pp. 417–418, 2009.
[17] O. Cordón, R. Alcalá, J. Alcalá-Fdez, and I. Rojas, "Special Issue on Genetic Fuzzy Systems: What's Next?" *Editorial, IEEE Transactions on Fuzzy Systems*, vol. 15, no. 4, pp. 533–535, 2007.
[18] H. Isibuchi, "Multiobjective genetic fuzzy systems: review and future research directions," in *Proceedings of the 2007 IEEE international conference on fuzzy systems (FUZZ-IEEE'07)*, London, 2007, pp. 913–918.
[19] T. P. Hong, K. Y. Lin, and S. L. Wang, "Fuzzy data mining for interesting generalized association rules," *Fuzzy sets and systems*, vol. 138, pp. 255–269, 2003.
[20] T. P. Hong, C. H. Chen, Y. L. Wu, and Y. C. Lee, "A GA-based fuzzy mining approach to achieve a trade-off between number of rules and suitability of membership fuctions," *Soft Computing*, vol. 10, no. 11, pp. 1091–1101, 2006.
[21] M. Kaya, "Multi-objective genetic algorithm based approaches for mining optimized fuzzy association rules," *Soft Computing*, vol. 10, no. 7, pp. 578–586, 2006.
[22] C. H. Tsang, S. Kwong, and H. Wang, "Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection," *Pattern Recognition*, vol. 40, no. 9, pp. 2373–2391, 2007.
[23] C. Romero, P. González, S. Ventura, M. J. del Jesus, and F. Herrera, "Evolutionary algorithm for subgroup discovery in e-learning: A practical application using Moodle data," *Expert Systems with Applications*, vol. 36, pp. 1632–1644, 2009.
[24] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimisation," in *Evolutionary methods for design, optimisation and control*, ser. CIMNE, e. a. K. Giannakoglou, Ed., 2002, pp. 95–100.
[25] K. Deb, A. Pratap, S. Agrawal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
[26] B. L. Miller and D. E. Goldberg, "Genetic Algorithms, Tournament Selection, and the Effects of Noise," *Complex System*, vol. 9, pp. 193–212, 1995.
[27] J. H. Holland, "Adaptation in natural and artificial systems," *University of Michigan Press*, 1975.
[28] A. Asuncion and D. J. Newman, "UCI machine learning repository," 2007. [Online]. Available: http://www.ics.uci.edu/∼mlearn/MLRepository.html

TABLE I
RESULTS OF THE EXPERIMENTATION FOR DISCRETE DATA SETS WITH TWO CLASSES

$Algorithm$	$\sharp Rul$	$\sharp Var$	$SIGN$	$WRAcc$	SUP_cN	$FCNF$
Tic-tac-toe (9 discrete variables, 2 classes, 958 examples)						
NMEF-SD Can	1.00	2.00	5.240	0.069	0.584	0.799
NMEF-SD DNF	2.26	2.79	4.455	**0.077**	0.764	0.774
MESDIF Can	6.00	3.14	5.005	0.042	0.304	0.747
MESDIF DNF	7.72	3.14	4.929	0.045	0.406	0.721
SDIGA Can	7.42	3.86	6.084	0.030	0.194	**0.817**
SDIGA DNF	6.72	3.64	**6.133**	0.030	0.408	0.780
SDIGA-II Can	2.73	2.01	3.406	0.042	0.498	0.633
SDIGA-II DNF	2.25	2.00	0.556	0.002	**0.795**	0.516
Breast-w (9 discrete variables, 2 classes, 699 examples)						
NMEF-SD Can	2.90	2.38	**22.722**	0.162	0.846	**0.955**
NMEF-SD DNF	8.48	5.04	22.412	**0.174**	**0.943**	0.932
MESDIF Can	11.90	2.42	19.409	0.116	0.710	0.896
MESDIF DNF	18.90	3.38	16.987	0.116	0.746	0.870
SDIGA Can	2.42	2.36	18.046	0.124	0.715	0.890
SDIGA DNF	4.28	5.54	19.891	0.129	0.667	0.804
SDIGA-II Can	2.04	1.76	1.597	0.009	0.064	0.095
SDIGA-II DNF	3.38	5.13	19.961	0.155	0.933	0.826
Vote (16 discrete variables, 2 classes, 435 examples)						
NMEF-SD Can	1.10	2.05	**21.974**	**0.217**	**0.946**	0.979
NMEF-SD DNF	2.22	2.95	21.884	**0.217**	**0.946**	**0.980**
MESDIF Can	7.86	3.44	19.937	0.187	0.827	0.957
MESDIF DNF	13.40	3.45	17.968	0.170	0.788	0.927
SDIGA Can	3.06	3.19	18.243	0.180	0.802	0.891
SDIGA DNF	2.28	2.17	20.335	0.208	0.931	0.923
SDIGA-II Can	2.93	2.27	18.843	0.199	0.920	0.905
SDIGA-II DNF	2.93	2.25	18.525	0.198	0.919	0.903

TABLE II
RESULTS OF THE EXPERIMENTATION FOR CONTINUOS DATA SETS WITH TWO CLASSES

$Algorithm$	$\sharp Rul$	$\sharp Var$	$SIGN$	$WRAcc$	SUP_cN	$FCNF$
Ion (34 continuous variables, 2 classes, 351 examples)						
NMEF-SD Can	8. 72	4.01	**7.513**	**0.144**	**0.966**	**0.879**
NMEF-SD DNF	11.16	5.04	1.101	0.012	0.474	0.761
MESDIF Can	19.74	5.26	3.809	0.056	0.638	0.734
MESDIF DNF	16.52	5.42	3.116	0.043	0.638	0.785
SDIGA Can	3.60	4.84	2.769	0.036	0.367	0.662
SDIGA DNF	8.34	5.08	2.553	0.029	0.266	0.649
SDIGA-II Can	2.08	2.11	1.612	0.029	0.298	0.298
SDIGA-II DNF	2.01	5.01	6.662	0.099	0.955	0.700
Haberman (3 continuous variables, 2 classes, 306 examples)						
NMEF-SD Can	1.00	2.00	0.767	**0.050**	0.933	**0.803**
NMEF-SD DNF	14.12	2.87	0.580	0.006	0.659	0.746
MESDIF Can	18.10	3.05	0.721	0.013	0.525	0.569
MESDIF DNF	7.46	2.49	0.719	0.015	0.739	0.558
SDIGA Can	2.00	2.00	**1.258**	0.042	0.837	0.635
SDIGA DNF	2.10	3.08	0.733	0.022	0.965	0.564
SDIGA-II Can	2.12	2.00	0.792	0.018	0.796	0.541
SDIGA-II DNF	2.00	3.13	0.395	0.022	**0.991**	0.521
Heart (13 continuous variables, 2 classes, 270 examples)						
NMEF-SD Can	4.10	2.61	**3.622**	**0.104**	0.769	**0.777**
NMEF-SD DNF	16.10	4.24	2.680	0.055	0.478	0.750
MESDIF Can	20.00	3.58	3.068	0.058	0.584	0.775
MESDIF DNF	19.84	3.94	3.117	0.062	0.454	0.774
SDIGA Can	2.00	2.08	2.426	0.078	0.678	0.628
SDIGA DNF	2.00	3.36	2.426	0.083	**0.968**	0.601
SDIGA-II Can	2.24	2.12	1.317	0.056	0.888	0.596
SDIGA-II DNF	2.00	3.92	3.271	0.092	0.957	0.611

TABLE III
RESULTS OF THE EXPERIMENTATION FOR DISCRETE DATA SETS WITH MORE THAN TWO CLASSES

$Algorithm$	$\sharp Rul$	$\sharp Var$	$SIGN$	$WRAcc$	SUP_cN	$FCNF$
Car (6 discrete variables, 4 classes, 1728 examples)						
NMEF-SD Can	1.10	2.00	**37.848**	**0.092**	0.439	**1.000**
NMEF-SD DNF	3.46	2.52	25.569	0.082	0.606	0.902
MESDIF Can	10.50	3.34	13.511	0.026	0.353	0.308
MESDIF DNF	25.68	4.34	22.238	0.039	0.509	0.568
SDIGA Can	16.80	5.03	1.935	0.002	0.048	0.238
SDIGA DNF	4.04	3.88	33.018	0.045	0.703	0.413
SDIGA-II Can	5.21	2.00	20.708	0.048	0.590	0.490
SDIGA-II DNF	4.53	3.79	33.468	0.055	**0.919**	0.594
Dermatology (33 discrete variables, 6 classes, 366 examples)						
NMEF-SD Can	2.06	6.38	**23.688**	**0.199**	**0.986**	**0.934**
NMEF-SD DNF	8.68	6.14	16.486	0.119	0.849	0.920
MESDIF Can	29.96	9.64	15.404	0.098	0.802	0.794
MESDIF DNF	23.44	5.24	11.415	0.064	0.709	0.540
SDIGA Can	6.00	.2.02	0.119	0.000	0.002	0.009
SDIGA DNF	6.00	1.95	0.232	0.000	0.001	0.003
SDIGA-II Can	6.00	1.93	0.171	0.000	0.001	0.008
SDIGA-II DNF	6.00	1.94	0.144	0.000	0.000	0.001
Lymp (18 discrete variables, 4 classes, 148 examples)						
NMEF-SD Can	11.38	3.72	**3.238**	**0.094**	0.516	0.630
NMEF-SD DNF	24.24	4.78	2.276	0.092	**0.716**	**0.633**
MESDIF Can	38.68	4.78	1.516	0.045	0.343	0.401
MESDIF DNF	19.84	3.35	1.802	0.058	0.529	0.564
SDIGA Can	4.02	1.84	0.149	0.004	0.078	0.071
SDIGA DNF	6.94	4.63	1.632	0.048	0.307	0.397
SDIGA-II Can	4.00	1.89	0.143	0.003	0.089	0.058
SDIGA-II DNF	4.80	3.73	0.755	0.024	0.466	0.302

TABLE IV
RESULTS OF THE EXPERIMENTATION FOR CONTINUOS DATA SETS WITH MORE THAN TWO CLASSES

$Algorithm$	$\sharp Rul$	$\sharp Var$	$SIGN$	$WRAcc$	SUP_cN	$FCNF$
Led (7 continuous variables, 10 classes, 500 examples)						
NMEF-SD Can	4.70	3.37	17.227	**0.064**	0.786	0.624
NMEF-SD DNF	30.58	6.23	12.772	0.048	0.553	**0.768**
MESDIF Can	78.54	3.56	17.006	0.045	0.818	0.377
MESDIF DNF	30.80	2.90	14.598	0.031	0.890	0.225
SDIGA Can	10.04	4.55	15.998	0.058	0.713	0.720
SDIGA DNF	12.10	4.50	15.196	0.057	0.731	0.728
SDIGA-II Can	10.00	2.01	**18.423**	0.036	0.908	0.199
SDIGA-II DNF	10.00	2.00	0.931	0.002	**0.988**	0.110
Cleveland (13 continuous variables, 5 classes, 303 examples)						
NMEF-SD Can	1.40	3.00	**10.034**	**0.135**	0.681	**0.860**
NMEF-SD DNF	7.90	4.11	4.268	0.053	0.487	0.739
MESDIF Can	48.26	4.48	3.951	0.020	0.496	0.277
MESDIF DNF	17.08	3.19	3.408	0.016	0.685	0.251
SDIGA Can	5.18	2.16	0.559	0.007	0.093	0.079
SDIGA DNF	17.32	6.03	1.012	0.007	0.117	0.137
SDIGA-II Can	5.02	1.56	0.775	0.006	0.218	0.083
SDIGA-II DNF	5.00	6.49	5.185	0.032	**0.883**	0.223
Glass (9 continuous variables, 6 classes, 214 examples)						
NMEF-SD Can	3.94	3.89	3.739	**0.035**	0.571	**0.821**
NMEF-SD DNF	7.28	4.24	1.825	0.011	0.289	0.432
MESDIF Can	18.00	4.68	1.644	0.005	0.311	0.233
MESDIF DNF	21.64	4.39	3.289	0.016	0.512	0.326
SDIGA Can	9.90	3.70	1.740	0.007	0.282	0.253
SDIGA DNF	8.72	6.83	1.436	0.008	0.357	0.326
SDIGA-II Can	6.00	2.98	2.696	0.007	0.516	0.197
SDIGA-II DNF	6.00	6.41	**4.014**	0.022	**0.691**	0.341