Information Sciences 298 (2015) 180–197

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A fuzzy genetic programming-based algorithm for subgroup discovery and the application to one problem of pathogenesis of acute sore throat conditions in humans

C.J. Carmona [a,*], V. Ruiz-Rodado [b], M.J. del Jesus [c], A. Weber [d], M. Grootveld [b], P. González [c], D. Elizondo [e]

[a] Department of Civil Engineering, Languages and Systems Area, University of Burgos, 09001 Burgos, Spain
[b] Leicester School of Pharmacy, De Montfort University, LEI 9BH Leicester, United Kingdom
[c] Department of Computer Science, University of Jaén, 23071 Jaén, Spain
[d] Institute for Materials Research and Innovation, University of Bolton, BL3 5AB Bolton, United Kingdom
[e] School of Computer Science and Informatics, De Montfort University, LEI 9BH Leicester, United Kingdom

ARTICLE INFO

Article history:
Received 24 May 2014
Received in revised form 19 November 2014
Accepted 21 November 2014
Available online 5 December 2014

Keywords:
Genetic programming
Subgroup discovery
Evolutionary fuzzy system
Bioinformatics

ABSTRACT

This paper proposes a novel algorithm for subgroup discovery task based on genetic programming and fuzzy logic called Fuzzy Genetic Programming-based for Subgroup Discovery (FuGePSD). The genetic programming allows to learn compact expressions with the main objective to obtain rules for describing simple, interesting and interpretable subgroups. This algorithm incorporates specific operators in the search process to promote the diversity between the individuals. The evolutionary scheme of FuGePSD is codified through the genetic cooperative-competitive approach promoting the competition and cooperation between the individuals of the population in order to find out the optimal solutions for the SD task.

FuGePSD displays its potential with high-quality results in a wide experimental study performed with respect to others evolutionary algorithms for subgroup discovery. Moreover, the quality of this proposal is applied to a case study related to acute sore throat problems.

1. Introduction

Subgroup discovery (SD) is a descriptive data mining technique for describing unusual features with monitored properties of interest [40,66]. This task contributes interesting knowledge to the scientific community from two view-points, specifically both features those including the provision of interest and precision. SD has been included within the concept of Supervised Descriptive Rule Discovery [42], together with further descriptive techniques such as emerging patterns [18] and contrast set mining [5].

Differing SD algorithms have been implemented throughout the literature in order to solve SD tasks based on beam search such as CN2-SD [45] or SD [27], exhaustive such as SD-Map [3], or genetic algorithms such as SDIGA [15] and NMEEF-SD [8], amongst others.

* Corresponding author. Fax: +34 259478.
E-mail address: cjcarmona@ubu.es (C.J. Carmona).

http://dx.doi.org/10.1016/j.ins.2014.11.030
0020-0255/

Genetic programming [41] is a methodology based on evolutionary algorithms (EAs) and it has been used for classification purposes [20], rule learning [43,44] and genetic-based machine learning [21]. Amongst its advantages can be highlighted the following:

- flexibility in the learning process due to the use of populations with dynamic size and individuals with structure and size variable. This property facilitates the obtaining of descriptive rules for the search space,
- simplicity, since it allows to learn rules in a flexible way without the necessity to include all variables in the individuals,
- diversity amongst the rules, which is acquired through specific operators promoting the diversity at phenotype or genotype level.

This paper presents a new approach named Fuzzy Genetic Programming-based for Subgroup Discovery: FuGePSD. This algorithm represents an evolutionary fuzzy system (EFS) [34] based on genetic programming [41] which employs a tree structure with a variable-length to represent the individuals of the population. FuGePSD employs several genetic operators in order to obtain rules to which are as general and precise as possible describing new information of the search space. In this way, FuGePSD includes an operator to promote the diversity at genotype level where rules describing the same examples are penalised. Moreover, drop and the insertion of genetic operators enhances the increase in precision and generality of the rules.

Benefits offered by the FuGePSD technique are delivered in a complete experimental study supported by appropriate statistical tests. The study is focused on datasets with continuous variables and the validity of FuGePSD is analysed with respect to alternative EAs for SD. Statistical tests confirm the highly effective performance and suitability for this new approach. Moreover, the behaviour of FuGePSD in real problems is applied to a study related to sore throat. This problem is an acute upper respiratory tract infection that impinges on the throat's respiratory mucosa, and can be linked with fever, headache and general malaise. The dataset analysed distinguishes for the high dimensionality with a wide number of features. Results acquired show the quality of the new proposal presented in this paper which are highlighted by experts in this field.

The paper is organised as follows. Firstly, preliminary concepts are described in Section 2. Next, Section 3 presents the new approach in which a description of the algorithm, operation scheme, fitness functions and genetic operators required in order to facilitate its analysis can be observed. Sections 4 and 5 present all information related to the experimental framework and the study, respectively. In Section 6, a case study is presented, and results arising there from are discussed by researchers with expertise in this field. Finally, the major salient conclusions are outlined.

2. Preliminaries

This section introduces the main concepts used for the algorithm presented. Firstly, a brief introduction to EFSs, and a short review of the SD proposals based on EAs in the specialised literature are presented in Section 2.1. Secondly, the definition, main properties and elements of the SD technique are outlined in Section 2.2. Thirdly, major properties and quality measures for fuzzy rules in SD are summarised in 2.3. Finally, the use of EFSs in SD throughout the literature is presented in Section 2.4.

2.1. Evolutionary fuzzy systems

An EFS [34] can be described as a fuzzy system [68] augmented with a learning process based on evolutionary computation [19], such as those involving genetic algorithms [30,36], genetic programming [41], evolutionary programming [23] or evolution strategies [55], amongst others.

Fuzzy systems are usually considered in the form of fuzzy-rule based systems (FRBSs), which are composed of "IF–THEN" rules where both the antecedent and consequent can contain fuzzy logic statements. This simple and interpretable representation facilitates their application in a wide range of real-world problems such as the pioneering problems in control [52], modelling [53], classification [39]. On the other hand, the EAs are well known and widely used global search technique with the ability to explore a large search space such as regression [26], association rule mining [50] or instance selection [17], for example. Therefore, EAs can be used in the development of FRBSs offering much potential as a search tool, allowing the inclusion of domain knowledge and the obtaining of better rules.

Different schemes of representation for the EAs are considered within EFS:

1. "*Chromosome = rule*" approach, in which each individual codifies a single rule, and the whole rule set is provided by combining several individuals within the population. Three categories are considered: the Michigan approach usually known as the as learning classifier system [37], the Iterative Rule-Learning approach [60] and the genetic cooperative-competitive learning approach [31].
2. "*Chromosome = set of rules*" approach, also known as the Pittsburgh approach, in which each individual represents a set of rules [59]. In this case, a chromosome evolves a complete set of rules that compete amongst them along the evolutionary process.

The redundancy in the evolutionary process for learning rules has been widely studied throughout the literature. There are proposals able to reduce the redundancy through specific operators as fusion of subsumptive rules for instance [61,62]. Another approach, for example, is represented through the token competition [48] which is used and explained in our method.

2.2. Subgroup discovery

SD task extracts knowledge in a descriptive manner from data concerning a property of interest. The concept of SD was initially introduced by Kloesgen [40] and Wrobel [66], and more formally defined by Siebes [57], but with use of the term Data Surveying for the discovery of interesting subgroups. It can be defined as in [67]:

> *"In subgroup discovery, we assume we are given a so-called population of individuals (objects, customers, ...) and a property of those individuals we are interested in. The task of subgroup discovery is then to discover the subgroups of the population that are statistically "most interesting", i.e., are as large as possible and have the most unusual statistical (distributional) characteristics with respect to the property of interest."*

The main purpose of SD is to seek and explore relationships between different properties or variables with respect to a target variable, and representations of the knowledge is performed through rules which consist of induced subgroup descriptions [27,45]. Each rule R can be formally defined as:

$$R : Cond \rightarrow Target_{value}$$

where $Target_{value}$ is a value for the variable of interest (target variable) for the SD task (which also appears as *Class* in the literature), and *Cond* is commonly a conjunction of features (attribute–value pairs) which is able to describe an unusual statistical distribution with respect to the $Target_{value}$. SD is a descriptive induction using supervised learning while classification is framed in predictive category. Main differences between SD and classification can be observed in [35].

Differing elements can be considered as the most important when an SD approach is applied. These elements are the type of the target variable, the search strategy that has an exponential relation to the number of features and the values considered, together with the descriptive language of the subgroups and the quality measures [4]. A review about major properties, features, algorithms and real-world problems solved through the application of SD algorithms can be founded in [35].

2.3. Fuzzy rules and quality measures for subgroup discovery

Fuzzy rules, based on fuzzy logic [68], allow already to consider uncertainty, and also to represent the continuous variables in a manner which is close to human reasoning. In this way, interpretable fuzzy rules consider continuous variables as linguistic where values are represented through fuzzy linguistic labels (*LLs*).

Eq. (1) represents a canonical fuzzy rule:

$$R : IF\ X_1 = (LL_1^2)\ AND\ X_3 = (LL_3^1)\ THEN\ Target_{value} \tag{1}$$

where:

- $X = \{X_m / m = 1, \ldots, n_v\}$ is a set of features used to describe the subgroups, and n_v is the number of descriptive features.
- $T = \{Target_{value} / j = 1, \ldots, n_{tv}\}$ is a set of values for the target variable, and n_{tv} is the number of values for the target variable.
- $LL_{n_v}^{l_{n_v}}$ is the *LL* number l_{n_v} of the variable n_v.

It is important to note that this type of rules are highly suitable for SD task. The fuzzy set corresponding to each *LL* can be specified by the user or defined by means of uniform partitions (if expert knowledge is not available).

The analysis of the quality of this type of rules is performed through different measures [35]. These quality measures presented in the following describe the ones used by:

- *Unusualness*: The weighted relative accuracy of a rule [46] measures interest and a trade-off between generality and precision. It can be computed as:

$$Unus(R_i) = \frac{n(Cond)}{n_s}\left(\frac{n(Target_{value} \cdot Cond)}{n(Cond)} - \frac{n(Target_{value})}{n_s}\right) \tag{2}$$

  It can be described as the balance between the coverage of the rule $p(Cond_i)$ and its accuracy gain $p(Target_{value} \cdot Cond) - p(Target_{value})$, where $n(Cond)$ is the number of examples which satisfy the conditions determined by the antecedent part of the rule, n_s is the number of total examples, $n(Target_{value} \cdot Cond)$ is the number of examples which satisfy the conditions and also belong to the value for the target variable within the rule, and $n(Target_{value})$ are all the examples of the target variable. The domain of this quality measure is specified for each

problem because there is a direct dependence with respect to the target variable as can be observed in the equation. Specifically, the domain is related to the percentage of instances of the majority target value ($\%majority_{Target_v}$) directly. In this way, bounds of unusualness are calculated as follows:

- Lower bound: $(1 - \%majority_{Target_v}) * (0 - \%majority_{Target_v})$
- Upper bound: $\%majority_{Target_v} * (1 - \%majority_{Target_v})$

- *Sensitivity*: This measure is the proportion of actual matches that have been classified correctly [40] and it has a component based on generality. It is computed as:

$$Sens(R_i) = \frac{n(Target_{value} \cdot Cond)}{n(Target_{value})} \tag{3}$$

This quality measures can be found in the literature as the Support based on the examples of the class, Recall or *TPrate*, and its domain is $[0, 1]$.

- *Fuzzy confidence*: This measures the relative frequency of examples satisfying the complete rule amongst those satisfying only the antecedent for fuzzy rules [15]. This is an adaptation of the standard confidence measure [1]. It is computed as:

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

An example E^k verifies the *APC* of a rule if

$$APC(E^k, R_i) = T(\mu_{LL_1^1}(e_1^k), \ldots, \mu_{LL_{n_v}^{l_{n_v}}}(e_{n_v}^k)) > 0 \tag{5}$$

Confidence measures precision within the domain $[0, 1]$, where *APC* (Antecedent Part Compatibility) is the degree of compatibility between an example and the antecedent component 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, where:

- $\mu_{LL_{n_v}^{l_{n_v}}}(e_{n_v}^k)$ is the degree of membership for the value of the feature n_v for the example E^k to the fuzzy set corresponding to the *LL*;
- T is the $t - norm$ selected to represent the meaning of the AND operator (the fuzzy intersection), in our case the minimum $t - norm$.

2.4. Evolutionary algorithms in subgroup discovery

Throughout the literature different EAs for SD have been presented [9]. All of them use EAs for the search process, and are able to obtain models which are both simple and precise. Subsequently, most relevant proposals within the literature are summarised.

- SDIGA [15] uses fuzzy rules as knowledge representations and a mono-objective EA as a learning process. SDIGA follows the IRL approach, where the solution of each iteration is the best individual obtained, and the global solution is formed via the best individuals obtained by the differing runs. SDIGA is executed for each value of the target variable, and it is interesting to remark that it always obtains rules for all classes of the dataset.
  The fitness function of the EA serves as an aggregation function where the selection of the quality measures such as significance, unusualness, sensitivity, support or confidence, amongst others, is determined by the user, where the number of objectives within the weighted aggregation function are between 1 and 3. The final rules obtained for each run are improved in a post-processing phase throughout a hill-climbing process, which modifies the rule in order to increase the degree of support.
- NMEEFSD [8] is a multi-objective EA for extracting fuzzy rules. This algorithm is based on the NSGA-II algorithm [14], and it allows researchers to choose between two or three quality measures as objectives of the evolutionary process in order to obtain relevant subgroups, i.e. those between coverage, significance, unusualness, accuracy, sensitivity, support and/or confidence. In NMEEF-SD, each candidate solution is coded according to the genetic cooperative competitive learning approach [31] which is within the "*chromosome = rule*" approach.
  In NMEEFSD, only the antecedent is represented in the chromosome, and the consequent is prefixed to one of the possible values of the target feature in the evolution. Therefore, the algorithm must be executed as many times as the number of different values that the target variable contains. In the final phase of the evolutionary process, NMEEFSD obtains only rules which reach a determined threshold of confidence.
- CGBA-SD [49] is called comprehensible grammar-based algorithm for subgroup discovery and combines the requirements of discovering comprehensible rules with the ability to mine expressive and flexible solutions owing to the use of a context free grammar. In this way, the algorithm employs the genetic programming paradigm [41] and genotype is defined by means of a tree structure with different shapes and sizes, i.e. CGBA-SD is within the "*chromosome = rule*" approach. This algorithm obtains crisp rules with intervals for continuous variables.

The initialisation of the population always generates individuals with fitness over 0 and these individuals are evaluated with a lineal fitness function calculated through sensitivity and confidence. This algorithm employs intervals generated in a random way for real domains and concrete values for the remainin type of variables. In addition, it is able to change the probability values used by the genetic operators. Finally, individuals of the final population are deleted with respect to a minimum confidence threshold and equivalence between individuals.

The ability of EAs have been demonstrated in several real-world problems with respect to SD in differing fields. Below, we outline a summary regarding the recent successes of types of systems in SD:

- *Bioinformatics* [7]. SD was applied in order to search for similarities and differences between different subtypes of Influenza A virus, and in this study proteins were extracted from signal processing techniques. The results acquired can serve to facilitate our understanding of these proteins in the pharmacological industry, and the fuzzy rules obtained were able to describe all viruses with high quality results, and using only a subset of the variables out of total available.
- *Concentrating photovoltaics* [11]. SD was applied in a problem related to the descriptive behaviour of a type of solar cell in concentrating photovoltaics. This technology serves as a cheaper alternative than conventional photovoltaics for electric generation processes. The main objective was to obtain subgroups able to provide new information to experts involved the maximum module power. Specifically, the results obtained have shown the necessity to analyse the influence of the APE in performance of the modules involved.
- *Web usage mining* [12]. An SD algorithm was applied to a real-world dataset extracted from the website related to the extra virgin olive oil www.orolivesur.com. The major objective was to analyse and extract valuable information from the website with data acquired using Google Analytics. The knowledge extracted offered different recommendations to the webmaster team related to the design and references website.
- *Medicine* [10]. This paper involved an analysis of the type of patients who tend to visit a psychiatric emergency department in a given period of time of the day was analysed. The main objective was to obtain subgroups in order to describe patients according to their time of arrival at the emergency department. Fuzzy subgroups extracted offered much useful knowledge to the experts regarding the distribution and allocation of medical resources at the hospital.
- *Marketing* [15]. In this paper the main objective was to arrive at conclusions from information available on previous trade fairs in order to determine relationships between the trade fair planning variables and the success of the stands involved. Knowledge extracted therefore has allowed the experts to obtain novel conclusions concerning data available in order to improve the organisation of new fairs of this nature.

3. FuGePSD: Fuzzy Genetic Programming-based learning for Subgroup Discovery

This section presents the approach for SD called Fuzzy Genetic Programming-based learning for Subgroup Discovery, FuGePSD. It involves an EFS based on a genetic programming algorithm [41] with the ability to extract descriptive fuzzy rules for the SD task.

The following subsections present the main concepts of FuGePSD. A complete description of the algorithm, its components and scheme are delineated in Section 3.1. Secondly, a representation of the fuzzy individuals through the context-free grammar definition route is outlined in Section 3.2. Thirdly, Section 3.3 presents the fitness functions employed in this approach. In Section 3.4 the most important operator of the algorithm is observable (the token competition). Finally, Section 3.5 describes the genetic operators used by the algorithm.

3.1. General scheme of operation for FuGePSD

The algorithm commences from an initial population generated in a random manner where individuals are represented through the "chromosome = individual" approach including both the antecedent and the consequent of the rule. Specifically, FuGePSD employs the genetic cooperative-competition approach where rules of the population cooperate and compete between them in order to obtain the optimal solution. The inclusion of the target variable in the representation is often an advantage with respect to alternative EAs available for SD, since while FuGePSD is executed only once obtaining rules for the different values of the target variable. However, the remaining proposals based on EAs are required to be executed once for each value of the target variable. It is also important to remark that individuals have variable length just like population with a variable number of individuals throughout the evolutionary process. In this way, FuGePSD is able to obtain rules with different number of variables in the antecedent part of the rule associated to the complexity of the subgroup to describe. On the other hand, the initial number of rules generated for the problem is adapted throughout the evolutionary process with respect to the problem to solve through different operators.

FuGePSD evolves with the generation of offspring populations through the application of several genetic operators. This population is generated with the same size than the parent population with respect to the number of individuals. Both populations are joined in a new population, in which the token competition operator is applied. This operator is crucial to the functioning of the algorithm in order to obtain diverse subgroups.

As can be observed in pseudo code of the Algorithm 1, this evolutionary process is controlled through the number of generations. It is important to note that for each generation, both individuals and population are evaluated in a separate manner, since in view of the use of the cooperative-competitive approach it is necessary to evaluate the individuals and populations through two independent fitness functions. Hence, individuals compete between themselves with respect to a local fitness, and cooperate in order to obtain a population which is more adapted to the problem. Once the evolutionary process has finished, the algorithm performs a screening function to obtain rules only with values greater than a threshold of sensitivity and confidence. These thresholds can be modified through external parameters providing to the experts an algorithm more adaptable to complex problems. In general, these thresholds should be configured upper than 60% level because subgroups obtained must be precise and general and both quality measures are ideal to meet these objectives. With these values, we may secure the extraction of interesting and effective rules for the SD task presented. The screening function is applied in the best population (*BestPop*) obtained throughout the complete evolutionary process.

Algorithm 1. Operation pseudo code for the FuGePSD algorithm

```
Output
RuleSet
Begin
Generate MainPop
Evaluate MainPop
BestPop ⟵ MainPop
repeat
  Generate OffspringPop through GeneticOperators
  Evaluate OffspringPop
  Join MainPop and OffspringPop in JoinPop
  MainPop ⟵ TokenCompetition(JoinPop)
  if MainPop.Fitness > BestPop.Fitness then
    BestPop ⟵ MainPop
  end if
until Number of generations is reached
RuleSet = ScreeningFunction(BestPop)
End
```

3.2. Representation of individuals through the free grammar definition context

FuGePSD utilises a context-free grammar which allows the learning of fuzzy rules and the absence of some input features, a process giving rise to compact and simple rules. Table 1 represents grammar example for a SD task with two features (X_1, X_2), five *LLs* per feature ($LL_1^1, LL_1^2, \ldots, LL_1^5, LL_2^1, \ldots, LL_2^5$) and two values for the target variable (Tv_1, Tv_2) where the symbol $?a$ in some of the production rules of the grammar represents one, and only one, of the values separated by commas in the square brackets. The rules of this grammar are considered in order to generate the initial population of the FuGePSD algorithm. This population is completely generated in a random way where individuals contain a number of descriptors between 1 and the 50% of the variables of the problem.

With respect to the use of fuzzy logic, as previously mentioned, the fuzzy sets corresponding to the *LLs* can be specified by the user, or alternatively, defined by means of an uniform partition (if the expert knowledge is not available). Even though FuGePSD can use both representations, in this paper uniform partitions with triangular membership functions are employed as shown in Fig. 1.

3.3. Fitness functions

As we have mentioned previously, FuGePSD follows a genetic cooperative-competitive learning approach [31], i.e. FuGePSD encodes a single rule per individual and they compete and cooperate simultaneously. This makes it necessary to consider not only the characteristics of individual rules but also the cooperation amongst rules. To do so, FuGePSD requires the use of two different fitness functions in order to optimise the individuals of the population via a localised one, and also optimise the whole population through a global fitness function which evaluates the accuracy of the set of rules as a means for classification. Both functions will be referred to as fitness function and global fitness, respectively.

- Fitness function: This is calculated through a specific quality measure for SD task chosen by the experts. Specifically, the algorithm is able to work with one of the different quality measures presented in this paper: *Unusualness* (Eq. (2)), *Sensitivity* (Eq. (3)) and and *Fuzzy Confidence* (Eq. (4)). The use of one quality measure as objective in the evolutionary process

Table 1
Grammar example.

Start $\longrightarrow$ [If], antec, [then], target_variable, [·]
antec $\longrightarrow$ descriptor1, [and], descriptor2
descriptor1 $\longrightarrow$ [any]
descriptor1 $\longrightarrow$ [X_1] is label
descriptor2 $\longrightarrow$ [any]
descriptor2 $\longrightarrow$ [X_2] is label
label $\longrightarrow$ member(?a,[$LL^1, LL^2, LL^3, LL^4, LL^5$]), [?a]
target_variable $\longrightarrow$ [Target_value is] descriptor
descriptor $\longrightarrow$ member(?a,[Tv_1, Tv_2]), [?a]

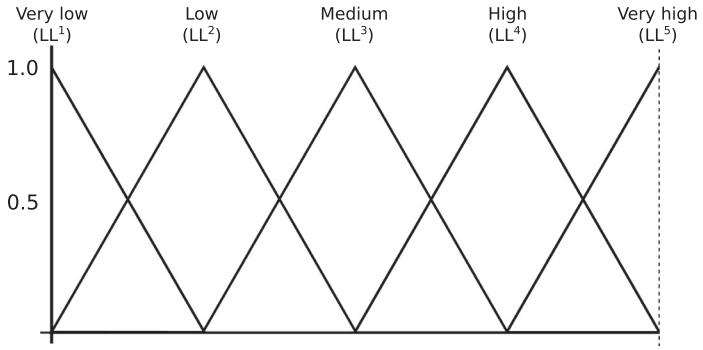


Fig. 1. Example of fuzzy partition for a continuous variable with five linguistic labels.

usually allows to the algorithm that the individuals with the best values in this quality measure for SD will be chosen and selected for evolution of the process. In general, a good behaviour for SD is achieved through the use of unusualness as objective which contributes with novelty and interest [45]. If the experts decide to use sensitivity in the evolutionary process, the subgroups obtained are more general with a low number of variables, whereas with the use of confidence it is possible to obtain the obtaining of very precise but less compact subgroups. The choice of the fitness function provides a major enhancement of the versatility to the algorithm, since experts could adapt the knowledge extraction process with respect to the nature of the problem. The fitness function is chosen through an external parameter of the algorithm.

- Global fitness: This is estimated through an adaptation score in order to obtain the best population during the whole evolutionary process. In this way, it is necessary to calculate the accuracy of the set of rules using the normalised sum of the predictions for each rule. The global fitness is defined as follows:

$$GlobalFitness = \frac{w_1 * Average\ Accuracy\ Rate + w_2 * (1 - n_v) + w_3 * (1 - n_R)}{w_1 + w_2 + w_3} \tag{6}$$

where n_R is the average number of rules for the population, n_v the number of descriptive variables, and *Average Accuracy Rate* [22] the mean value for the accuracy of each single value of the target variable (calculated as):

$$Average\ Accuracy\ Rate = \frac{1}{n_{tv}} * \sum_{i=1}^{n_{tv}} TPrate_i \tag{7}$$

Different weights (w_1, w_2 and w_3) are used in order to give a major 'trade-off' between accuracy and interpretability in order to satisfy all requirements of SD (general, precise and interest). These weights can be also modified through external parameters in order to facilitate the experts adaptability to real and complex problems. In this manner, it is recommendable to use values upper than 0.7 out of 1 for w_1, and the remaining 0.3 out of 1 for w_2 and w_3, because the main idea is to obtain a precise model with a low number of rules and a low number of variables for the set of rules. The use of $1n_v$ and $1n_R$ gives rise to an excessive number of rules and variables being penalised in the population score. With regard to the accuracy measure, the use of standard accuracy rate could lead to wrong conclusions since it does not consider the rate between classes. Therefore, the average accuracy rate is used since an equitable weight for all classes of the problem are independent of the number of examples for which each one is applied. The result of this accuracy rate is more suitable for obtaining an homogeneous accuracy for all values of the target variable.

3.4. Token competition

FuGePSD employs the token competition operator [48,63] in order to improve the diversity amongst the individuals at phenotipic level emulate the behaviour in a natural environment. Individuals with good niches will attempt to exploit that one alone and prevent further individuals to share its resources, unless a newer one is stronger that that initially developed. Therefore, the other individuals are required to seek their own niches. These properties provide to the algorithm diversity in the population. Moreover, this operator reduces the number of rules because all individuals without tokens will be deleted.

In our proposal, an example is considered as a token and all individuals in the population will compete for this example. When an individual matches one example provided, a flag will indicate that example is seized and hence other individuals cannot capture it. In this way, the diversity is improved with respect to the phenotype, i.e. individuals that overcome the token competition describe information about examples of the dataset which are not covered for others rules. The objective of this operator is to increment the description of the rules base about the dataset reducing, as far as possible, the redundancy.

The operation of this mechanism starts with the order (from high to lows) of the population with respect to the individual fitness. Subsequently, an individual with the highest fitness will exploit its niches by seizing as many tokens as it can. The other individuals entering the same niches will have their strength decreased, since they cannot compete with the stronger ones. This is achieved by introducing a penalisation to the fitness score of each individual, a limit which is based on the number of tokens which each individual has seized:

$$PenalizedFitness(R_i) = unusualness(R_i) * \frac{count(R_i)}{ideal(R_i)} \tag{8}$$

where, $count(R_i)$ is the number of tokens of the rule actually seized, and $ideal(R_i)$ is the total number of tokens that it can seize which is equivalent to the number of examples that the rule matches. If one rule seizes zero tokens, its fitness is modified to zero directly. On termination of the application of this mechanism, the size of the population is reduced with the individuals, where *PenalizedFitness* is greater than zero.

It should also be noted, that:

- this mechanism allows us to obtain a series of rules, all of which cover at least one example of the dataset not yet covered by other stronger rules, and
- the number of rules obtained is reduced, since individuals that describe information about examples described by other rules are eliminated.

3.5. Genetic operators

These operators are used in order to generate an offspring population from the main population. All individuals generated in the new population are a modification of one individual of the main population applying crossover, mutation, insertion or dropping. Each child is created by applying only one of the genetic operators through a probabilistic way. It is important to highlight that all new individuals must meet with the rules of the context-free grammar defined in Table 1.

Genetic operators are applied on one individual selected through a binary tournament selection process [51] from the main population. The operation of these operators, together with graphical descriptions, are developed in order to facilitate an understanding of the algorithm.

3.5.1. Crossover

Crossover is a genetic operator that combines two individuals within the main focus that the new individual may be better than both of the parents if it heritages part of the properties from each of them. It is necessary to select a second individual in order to apply this operator. In this paper, a component of the first parent is randomly selected and exchanged by another part, in the second one (also randomly selected), but under the constraint that the offspring produced must be valid according to our grammar.

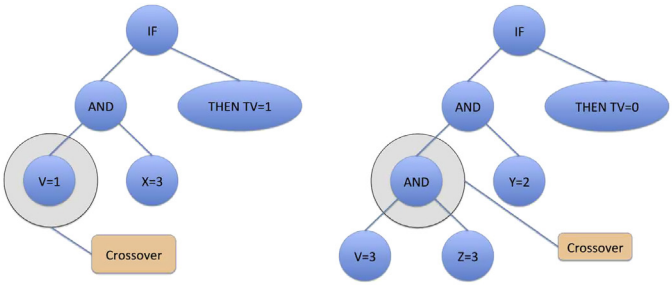
It should must be noted that the crossover operator in fact produces two children, but only one of them (randomly chosen) is returned as a descendant if this child is valid. If both children are invalid, the crossover operator is performed again.

Fig. 2(a) shows the initial individuals selected to cross. The results obtained of the application of this operator can be observed in Fig. 2(b) where the combination of both individuals are presented. As we have mentioned previously, a random process selects one children to include it in the offspring population.

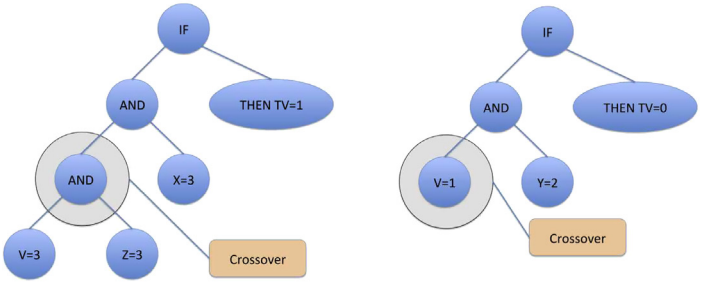
3.5.2. Mutation

This operator is used to improve the diversity from the initial to the offspring population. This operator alters one variable (selected in a random manner) from the individual with a value different to the original one considering the rules of the grammar.

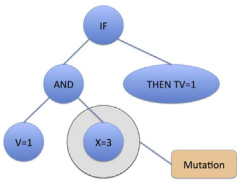
In Fig. 2(c) the mutation of the individual can be observed with a modification of the value for the variable *X*. FuGePSD only performs mutation with respect to the values of the variable.



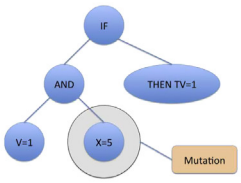
(a) Initial individuals for applying the crossover operator

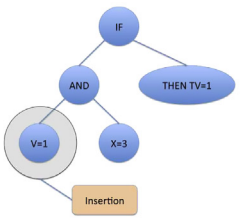


(b) Individuals obtained after the crossover

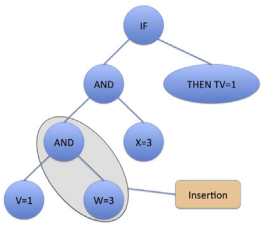


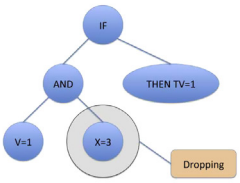
(c) Individual mutated

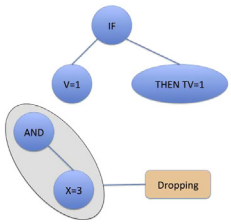




(d) Individual with an inserted node







(e) Individual with a dropped node

Fig. 2. Genetic operators of the algorithm FuGePSd.

3.5.3. Insertion

The insertion of variables in an individual purposes to include more precise rules in the model, with the objective of improving the precision and confidence of set of rules of the algorithm. This operator inserts a new variable in the individual with an associated value to the variable generated in a random manner. It is not applied if the individual already has all variables initialised according to the grammar. Fig. 2(d) represents the insertion operator for the FuGePSD algorithm in which variable *W* is included in the individual selected.

3.5.4. Dropping

This operator attempts to generate more generalised individuals which improve their support and sensitivity measures. In addition, in view of the probabilistic nature of genetic programming, redundant constraints may be generated in the individual. Thus, it is necessary to generalise the rules, in order to represent the knowledge in a more concise form.

The algorithm randomly selects one variable in the individual and it removes. This variable is no longer considered in the rule, and so it selects can be generalised. This operator is not applied if the individual hence has only one variable following with the grammar of the algorithm. Fig. 2(e) shows the dropping operator for the algorithm, where the variable X is removed from the initial individual.

3.5.5. Screening function

For the final extraction of the best rules, a screening function is applied. This function can be observed in Alg. 2. The screening is performed for the complete (best) population *BestPop* where the algorithm obtains the best rules for the dataset which must reach a determined values in confidence and sensitivity.

Next, function contains an external parameter (*AllTargetValues*) to indicate the type of rules extracted. There are two options for this parameter, first one, the *TRUE* value indicates the necessity of obtain rules for all values of the target variable. With this value, a check for each target value is analysed to include rules for all target values in the final rule set. On the other hand, for the *FALSE* value, the function includes the best rule of the full population if the final rule set is empty. In this way, the algorithm always obtains rules.

Algorithm 2. Screening function for the FuGePSD algorithm

```
Input
Population BestPop
Output
RuleSet RS
Begin
for i = 1 to i = BestPop.Size do
  R⟵getRule(BestPop, i)
  if R.getConf ⩾Confidence AND R.getSens ⩾Sensitivity then
    RS ∪ R
  end if
end for
if AllTargetValues = TRUE then
  for i = 1 to i = t_nv do
    if NumberRules(RS, i) = ϕ then
      R ⟵ BestPop.BestRule(t_i)
      RS ∪ R
    end if
  end for
else
  if NumberRules(RS) = ϕ then
    R ⟵ BestPop.BestRule()
    RS ∪ R
  end if
end if
End
```

The function $BestRule(t_i)$ and $BestRule()$ obtain the rule with the highest confidence value for a target variable and the full population, respectively. It is important to highlight that the screening function is applied on the best population, which is the population obtained with the best global fitness throughout the entire execution, i.e. best population has the best global fitness throughout the evolutionary process.

4. Experimental framework

This section outlines the main details of the experimental study performed. Specifically, Section 4.1 summarises the datasets analysed in the study for the SD algorithms presented (Section 4.2). Finally, Section 4.3 presents the statistical tests applied in order to analise the results obtained with respect to different EAs for SD task.

4.1. Datasets

The experimental study of this paper has been performed with 17 datasets obtained from the UCI Repository of machine learning databases [2]. The main characteristics of these datasets are summarised in Table 2 where ♯Vars are the number of

Table 2
Dataset characteristics.

Name	♯Vars	♯Disc	♯Cont	♯Cl	♯Inst	Name	♯Vars	♯Disc	♯Cont	♯Tv	♯Inst
Appendicitis	7	0	7	2	106	Haberman	3	0	3	2	306
Australian	14	8	6	2	690	Heart	13	6	7	2	270
Balance	4	0	4	3	625	Hepatitis	19	13	6	2	155
Bridges	7	4	3	2	102	Ionosphere	34	0	34	2	351
Cleveland	13	0	13	5	303	Iris	4	0	4	3	150
Diabetes	8	0	8	2	768	Led	7	0	7	10	500
Echo	6	1	5	2	131	Vehicle	18	0	18	4	846
German	20	13	7	2	1000	Wine	13	0	13	3	178
Glass	9	0	9	6	214						

total variables, ♯Disc and ♯Cont are the number of total discrete and continuous variables, respectively, ♯Cl are the number of values for the target variable and ♯Inst the number of instances of the dataset.

All datasets analysed in this experimental study have at least one continuous variable as can be observed in their main characteristics, i.e. an analysis in continuous environments is performed in this study since the major objective of this new proposal is focused on obtaining descriptive rules for these types of problems.

4.2. Algorithms

The validity of the proposal presented in this paper is shown with respect to different algorithms. Specifically, FuGePSD is compared with the most general EAs for SD available in the literature, SDIGA, NMEEF-SD and CGBA-SD. It is important to highlight that all datasets contain continuous variables and it is necessary to use algorithms able to work with this type of variables where a previous discretisation which could lead to a loss of precision in the results obtained. In addition, NMEEF-SD, SDIGA and CGBA-SD have confirmed the quality of results achievable with respect to those from other algorithms in [8,15,49], respectively. In fact, in these contributions these algorithms obtained better results than other ones of the literature between datasets used in this experimental study.

The parameters used for the algorithms in this experimental study are summarised in Table 3. These parameters have been selected with respect to the recommendations performed by the authors. On the other hand, FuGePSD employs the value false for the AllTargetValue parameter and rules are filtered with respect to sensitivity and fuzzy confidence. Estimation about quality measures are obtained through a 10-fold cross-validation. In addition, in view of the fact that all algorithms are stochastic, three executions are performed, and an average result from 30 values is shown for each dataset. It is important to highlight that values of a set of rules in unusualness, sensitivity and confidence are computed as the average for all rules.

4.3. Statistical tests for performance comparison

The use of statistical tests facilitates the analysis of experimental studies, since significant differences can be found amongst the differing algorithms employed for obtaining the superior behaviour of the one that achieves the highest average result. In view of the fact that the initial conditions that guarantee the reliability of parametric tests may not be satisfied (causing the statistical analysis to lose credibility with these type of tests) [16], non-parametric tests are employed in this paper.

The Friedman test [25] is used to compare the results obtained and to be able to precisely analyse whether there are significant differences amongst the four algorithms. This test first ranks the *jth* of *k* algorithms on the *ith* of *N* datasets, and then calculates the average rank according to the F-distribution (Distribution value) throughout all the datasets, and calculates the Friedman statistics. If the Friedman test rejects the null-hypothesis, indicating that there are significant differences. In this way, Holm test [38] is applied where the algorithm with the best result in this ranking is considered the control

Table 3
Parameters of the algorithms.

Algorithm	Parameters
SDIGA	Fitness = (0.7∗sensitivity + 0.3∗unusualness); linguistic labels = (3 and 5); minimum confidence = (0.6, 0.7, 0.8 and 0.9); population size = 100; maximum evaluations = 10,000; crossover probability = 0.60; mutation probability = 0.01
NMEEFSD	Objective1 = sensitivity; objective2 = unusualness; linguistic labels = (3 and 5); minimum confidence = (0.6, 0.7, 0.8 and 0.9); population size = 50; maximum evaluations = 10,000; crossover probability = 0.60; mutation probability = 0.10
CGBA-SD	Minimum confidence = (0.6, 0.7, 0.8 and 0.9); population size = 50; number of generations = 100
FuGePSD	Fitness = unusualness; linguistic labels = (3 and 5); minimum confidence = (0.6, 0.7, 0.8 and 0.9); minimum sensitivity = 0.6; population size = 200; maximum generations = 20,000; crossover probability = 0.50; mutation probability = 0.20; insertion probability = 0.15; dropping probability = 0.15; $w_1 = 0.7$; $w_2 = 0.15$; $w^3 = 0.15$; AllTargetValues = false

algorithm (Alg_{Cont}), which controls the Holm test. In the result tables for the Holm test the algorithms are shown in descending order of z. Thus, by using the normal distribution we can obtain the corresponding $p - Value$ associated with each comparison and this can be compared with the associated α/i in the same row of the table to show whether the associated hypothesis of equal behaviour is rejected in favour of the best ranking algorithm or not.

In addition, in this study is employed the Wilcoxon signed-rank test [56] which is analogous to the paired t-test. This test is a non-parametric statistical procedure for the comparing of two algorithms, which allows the detection of significant differences between them. The operation of this test is based on the computation of the difference between the results arising from the application of both algorithms. With these differences, a ranking is generated with respect to their absolute values, (i.e. from smallest to largest), and average ranks are assigned in the case of ties. However, it is important to note that there are two values, R^+ which is the sum of ranks for the datasets on which the second algorithm outperformed the first, and R^- the sum of ranks for the opposite. Let T be the smallest of the sums, $T = min(R^+, R^-)$. If T is less than or equivalent to the value of the distribution of Wilcoxon for $N_{datasets}$ degrees of freedom, the null hypothesis of equality of means is rejected. A hypothesis of comparison is rejected with respect to a specified level of significance. This level represents the lowest level of significance of a hypothesis that gives rise to a rejection. The significance of the p-value is determined as in [56].

Different studies in the literature suggest the use of non-parametrical tests in the field of machine learning [16,28,29], and additional information can be found on the Website http://sci2s.ugr.es/sicidm/.

5. Experimental study

This section presents results for each quality measures with respect to the algorithms considered in the study. NMEEFSD is abbreviated to NMEEF, CGBA-SD to CGBA and FuGePSD to FuGeP. An analysis of the best parameters (minimum confidence and number of linguistic labels) has previously been carried out. Results shown in Table 4 are the average results obtained with them.

FuGePSD obtains the highest values in the AVERAGE of the three quality measures analysed in this paper. However, the results are analysed for each quality measure below:

- Unusualness. This quality measure indicates the novelty or interest of the subgroups. The algorithm FuGePSD gets the best average result, and it attains the best results in 10 out of 17 datasets. Moreover, it is interesting to note that values obtained in datasets such as Ionosphere, Iris or Wine are very close to the maximal level.
- Sensitivity measures generality and precision in rules. Instead of FuGePSD does not obtain the best results in any datasets. However, it is interesting to note that the average value is the greatest with a value greater than 81%. In this way, FuGePSD obtains a good level of sensitivity in a homogeneous manner throughout the experimental study in opposite to the remaining algorithms.
- Confidence. This quality measure indicates the precision of the subgroups. As noted above, FuGePSD provides the best results with high quality with average values close to 90%. This algorithm obtains the best results in 10 out of 17 datasets.

In summary, the FuGePSD algorithm obtains the best average in this experimental study. However, despite this analysis it is necessary to compare results arising from the non-parametric Friedman and Holm tests in order to search for significant differences. In Table 5 results obtained in the Friedman test can be observed where distribution values are greater than the

Table 4
Detailed average results for each quality measure for the different algorithms analysed. The best case for each quality measure and dataset is highlighted in bold.

Unusualness					Sensitivity				Confidence			
Dataset	NMEEF	SDIGA	CGBA	FuGeP	NMEEF	SDIGA	CGBA	FuGeP	NMEEF	SDIGA	CGBA	FuGeP
Appendicitis	0.0945	**0.0966**	0.0640	0.0445	**1.0000**	0.9762	0.6670	0.9263	0.9038	0.7104	0.7170	**0.9529**
Australian	**0.1791**	0.0525	0.1770	0.1619	0.7989	**0.8736**	0.8550	0.8004	**0.9320**	0.6974	0.8580	0.8885
Balance	0.0710	0.0655	0.0720	**0.0815**	0.5259	0.5839	**0.6480**	0.6300	**0.6981**	0.5818	0.6180	0.6728
Bridges	0.0446	0.0322	0.0290	**0.0517**	**0.8787**	0.7076	0.6710	0.7231	0.9137	0.7345	0.7230	**0.9476**
Cleveland	**0.1300**	0.0155	0.1080	0.1153	**0.7763**	0.3577	0.7210	0.7701	**0.8232**	0.7442	0.7540	0.8138
Diabetes	**0.0859**	0.0651	0.0720	0.0822	0.9180	**0.9812**	0.5570	0.7038	0.7896	0.6226	0.7930	**0.8705**
Echo	0.0390	0.0333	0.0320	**0.0677**	0.8582	**0.9834**	0.4150	0.8090	0.7614	0.5705	0.6380	**0.8366**
German	**0.0669**	0.0020	0.0690	0.0323	0.4845	**0.8408**	0.4830	0.7911	0.8767	0.5677	**0.8800**	0.7434
Glass	0.0824	0.0176	0.0700	**0.0995**	0.8083	**0.9186**	0.5890	0.8931	0.8646	0.5501	0.6130	**0.9723**
Haberman	**0.0499**	0.0423	0.0230	0.0394	**0.9334**	0.8370	0.6610	0.9190	**0.8030**	0.6291	0.6420	0.7974
Heart	0.1072	0.0586	0.0820	**0.1318**	0.7175	**0.9706**	0.5410	0.7441	0.7650	0.5778	0.7560	**0.8729**
Hepatitis	0.0651	0.0352	0.0560	**0.0767**	**0.8059**	0.5845	0.6900	0.7425	0.8798	0.8269	0.6590	**0.9498**
Ionosphere	0.1413	0.0287	0.0310	**0.1888**	**0.9701**	0.8108	0.8480	0.9524	0.8663	0.5562	0.9050	**0.9679**
Iris	0.2067	0.1693	0.1050	**0.2098**	**1.0000**	0.9896	0.6720	0.9867	**0.9914**	0.8990	0.8510	0.9825
Led	0.0658	0.0593	**0.2040**	0.0769	0.8006	0.8172	**0.9530**	0.8548	0.6494	0.4948	**0.9430**	0.8255
Vehicle	0.0000	0.0240	0.0670	**0.0930**	0.0000	0.5962	**0.8400**	0.7079	0.0000	0.3070	0.6530	**0.7903**
Wine	0.1448	0.0862	0.0640	**0.1830**	**0.9190**	0.9071	0.4220	0.8819	0.8874	0.8927	0.6390	**0.9823**
Average	0.0926	0.0520	0.0858	**0.1021**	0.7762	0.8080	0.6573	**0.8139**	0.7885	0.6449	0.7451	**0.8745**

Table 5
Results of Friedman test in the quality measures analysed for the comparison of the algorithms.

Quality measure	Test value	Distribution value
Unusualness	6.25	21.00
Sensitivity	6.25	11.67
Confidence	6.25	33.20

Table 6
Results of Holm test to detail the results obtained for each quality measures for the comparison of the algorithms.

Quality measures	AlgControl	i	Algorithm	z	p	α/i	Hypothesis
Unusualness	FuGePSD	3	SDIGA	4.2602	2.0E−5	0.033	Rejected
		2	CGBA-SD	2.1946	0.0281	0.050	Rejected
		1	NMEEFSD	0.7745	0.4385	0.100	Non-rejected
Sensitivity	NMEEFSD	3	CGBA-SD	2.9692	0.0029	0.033	Rejected
		2	FuGePSD	0.5163	0.6055	0.050	Non-rejected
		1	SDIGA	0.1290	0.8972	0.100	Non-rejected
Confidence	FuGePSD	3	SDIGA	5.4221	5.8E−8	0.033	Rejected
		2	CGBA-SD	3.0983	0.0019	0.050	Rejected
		1	NMEEFSD	1.2909	0.1967	0.100	Non-rejected

Table 7
Results of the Wilcoxon test between NMEEFSD and FuGePSD.

Comparison	Quality measure	R^+	R^-	p-value	Hypothesis
FuGePSD vs. NMEEF-SD	Unusualness	118	35	0.049	Rejected by FuGePSD
	Sensitivity	97	56	0.332	Non-rejected
	Fuzzy confidence	108	28	0.039	Rejected by FuGePSD

test values. In this way the null-hypothesis that all algorithms perform equally well are rejected for all quality measures. Therefore, it is necessary to apply the Holm test in order to detail these differences between the algorithms.

Table 6 represents results of the Holm test where the algorithm with the best result in this ranking is considered the control algorithm (*AlgCont*) which controls the test. To analyse all hypothesis of the statistical studies we have used a level of significance of $\alpha = 0.10$.

The algorithm with the best ranking for unusualness and confidence is the proposal presented in this paper. In addition, in these quality measures FuGePSD obtains significant differences with respect to SDIGA and CGBA-SD because the null-hypothesis are rejected. However, in the sensitivity measure NMEEFSD obtains the best ranking with significant differences only with respect to CGBA-SD algorithm. In summary, algorithms with the best behaviour in this experimental study are NMEEFSD and FuGePSD. However, due to the absence of significance differences between both proposals is necessary to apply the Wilcoxon test (between both algorithms) in order to establish a ranking between them. Table 7 represents the results of this test where R^+ corresponds to the sum of the ranks for the first algorithm and R^- corresponds to the second algorithm.

Significant differences for unusualness and confidence in favour of the new proposal can be observed in Table 7, i.e. subgroups obtained for FuGePSD are more precise and with more interest. Moreover, Wilcoxon results for sensitivity quality measure shows that FuGepSD offers an improved ranking with respect to NMEEFSD, i.e. Wilcoxom confirms the previous conclusion with respect to sensitivity indicating that FuGePSD obtains the best level of sensitivity in an homogeneous manner throughout the experimental study though without significance differences.

It is clear that the FuGePSD algorithm obtains an improvement with regards those currently presented in SD, specially for EAs present in datasets with continuous variables. This new approach obtains subgroups which are more precise and with a higher novelty information.

6. A case study: pathogenesis of acute sore throat conditions in humans

Sore throat (sometimes known as 'pharyngitis' or 'tonsilitis') is an acute upper respiratory tract infection that impinges on the throat's respiratory mucosa, and can be linked with fever, headache and general malaise. Moreover, acute otitis media, acute sinusitis and peritonsillar abscess represent suppurative complications of this condition, predominantly the first of these. 85–95% of adult acute sore throat conditions are ascribable to viruses, as are 70% of those in children aged 5–16 years (and 95% of those in children aged < 5 years) [65]. However, the remainder arise from a bacterial source [predominantly group A β-hemolytic streptococcus (GABHS)]; clinically, the four most valuable features to identify in the diagnosis of sore throat diseases which are caused by GABHS are enlarged submandibular glands, the presence of a throat exudate and rhinorrhea (runny nose), together with the absence of fever and cough [54,65].

In this case study, we have coupled FuGePSD with high field proton (^{1}H) NMR spectroscopy analysis in order to recognise salivary biomolecule signatures which are characteristic of viral -(and, if applicable, bacterial)- induced acute sore throat conditions in humans. Specifically, healthy and clinically-diagnosed patients with acute sore throat conditions patients are analysed through FuGePSD in a problem with more than 200 variables and 500 instances. The main goal is to describe and characterise (from the point of view of SD) this problem with respect to the condition of the patients: healthy (control) and sore throat, i.e. with respect to two values for the target variable.

The applications of high field proton ^{1}H NMR spectroscopy to the detection and quantification of biomolecules present in complex biological fluids offers many advantages over other alternatives as can be observed in [13,32,33,58].

This case study has been performed in different stages: Firstly, patients were selected and data were collected as described in Section 6.1. Next, a preprocessing stage was applied to human saliva samples in order to obtain a data matrix for applying FuGePSD proposal in Section 6.2. Finally, Section 6.3 presents the analysis and results obtained with the FuGePSD algorithm.

6.1. Collection of human saliva samples

A series of patients with a clinically-diagnosed acute sore throat condition ($n = 50$) and healthy, non-medically-compromised age-matched controls ($n = 50$) were recruited to the study, the latter serving as essential controls. All of them were required to fully complete a participant questionnaire with both personal and medical information such as age, gender, body mass index, cough, rhinitis, fever history, etc. All participants were also instructed not to receive any form of medication during the 5-day trial. This investigation was performed by Professor Grootveld's research group, and full ethical approval for it was granted by the University of Bolton's Research Ethics Committee.

For the ^{1}H NMR data acquired we primarily implemented a rigorous analysis-of-variance (ANOVA)-based experimental design. This procedure was principally aimed at determining the significance of the 'Between-Disease Group' component of variance (and further ones involved) for the intensities of ^{1}H NMR intelligently-selected bucket signals which remained in the spectrum following the spectral editing process described below. A bucket is considered as an input variable in the dataset to analyse through the FuGePSD algorithm.

The experimental design selected was a combination of completely randomised with a randomised block design: mixed model with the 'Between-Participants' component of variance ($n = 50$ per Group) 'nested' within Disease Classification Group (Table 8). This model was preliminarily employed to probe the prognostic/diagnostic specificity of each 'Intelligently-Selected' Chemical Shift Bucket (ISB). Hence, this design allowed the study of each of these sources of variation simultaneously. For this ANOVA model, the complete dataset was log_{10}-transformed prior to analysis in order to satisfy assumptions of normality, variance homogeneity and additivity, etc. Saliva specimens were collected from each participant immediately after awakening in the morning as previously described in [58]. These 5 samplings took place throughout an intensive one-week period (Monday–Friday), and they were instructed to collect all saliva available in order to avoid interferences.

6.2. ^{1}H NMR analysis of human salivary supernatants

The preparation of human saliva samples for ^{1}H NMR analysis was performed as previously described [58]. Single-pulse ^{1}H NMR spectra of human salivary supernatant specimens were acquired on a Bruker Avance AM-600 spectrometer operating at a frequency of 600.13 MHz as described previously [47,58,64], as were both one- and two-dimensional ^{1}H–^{1}H COSY and TOCSY spectra.

Main objective in these stages was to obtain a ^{1}H NMR data matrix with a spectra for each saliva specimen with different input variables. In this way, a matrix with 500 spectra and 209 intelligent chemical shift buckets (ISB input variables) generated via the application of macro procedures for line-broadening, zero-filling, Fourier-transformation and phase and baseline corrections, followed by the application of a separate macro for the 'Intelligent Bucketing' processing sub-routine is obtained; all procedures were performed with the ACD/Labs 1D NMR Manager software package (ACD/Labs, Toronto, Canada M5C 1T4). These buckets are selected through the employment of an algorithm designed to make critical divisional decisions, i.e. those which define precisely the loci of bucket divisions with regard to an optimised selection of 'resonance-specific' ones (B. Lefebvre, Intelligent Bucketing for Metabonomics, ACD/Labs Technical Note, 2004). This strategy

Table 8
Experimental design for the analysis of each dataset of ^{1}H NMR ISB integration intensities, representing a combination of completely randomised with a randomised block design: mixed model with participants ($n = 50$ per group) 'nested' within each of the two disease classification groups.

Source of variation	Levels	Degrees of freedom	Nature	Parameters estimated
Between disease classifications	2	1	Fixed	$\sigma^2 + 5\sigma_{P(D)^2} + 250K_{D^2}$
Between participants	100	98	Random	$\sigma^2 + 5\sigma_{P(D)^2}$
Sampling days-withing-participants	5 per volunteer	4	Sequentially-fixed	$\sigma^2 + 100K_{S^2}$
Error (residual)	n/a	396	n/a	σ^2
Total	n/a	499	n/a	n/a

Table 9
Subgroups obtained in the case study through the FuGePSD method.

Sb		UNUS	SENS	FCNF
1	IF ISB (6.31–6.33) = Low AND ISB (0.60–0.62) = Medium AND ISB (1.36–1.40) = Medium AND ISB (3.55–3.61) = Medium AND ISB (6.83–6.88) = Medium THEN Control	0.0167	0.8917	0.6854
2	IF ISB (2.22–2.27) = Medium AND ISB (2.29–2.31) = Medium AND ISB (2.78–2.83) = Medium AND ISB (5.66–5.69) = Medium THEN Sore Throat	0.0260	0.6958	0.9772
3	IF ISB (2.22–2.27) = Medium AND ISB (2.78–2.83) = Medium AND ISB (5.66–5.69) = Medium THEN Sore Throat	0.0271	0.7000	0.9259
4	IF ISB (2.22–2.27) = Medium AND ISB (5.66–5.69) = Medium AND ISB (8.37–8.42) = Medium THEN Sore Throat	0.0448	0.6625	0.8859
5	IF ISB (2.22–2.27) = Medium AND ISB (5.66–5.69) = Medium THEN Sore Throat	0.0290	0.7292	0.8154

generated one global table of 'intelligently-selected bucket' (ISB) intensities. Chemical shift buckets containing less than 1% of the maximum summed intensity were removed from the dataset (since they may contain spectral 'noise').

After removal of the intense H_2O resonance (δ = 4.50–5.10 ppm), together with those arising from ethanol [centred at δ = 1.21 (t) and 3.66 ppm (q)], all ISB variables, or, where indicated were incorporated into the dataset for analysis with FuGePSD. All chemical shift bucket intensity values were normalised to that of the pre-added TSP internal standard (of fixed concentration).

6.3. Extraction of subgroup discovery by FuGePSD

Finally, the application of FuGePSD is performed on a dataset with 500 instances and 209 variables, and it is very important to note that buckets or input variables have a real domain. In this way, the use of this algorithm is relevant within SD task because as we have presented in the previous section, FuGePSD obtains the best results for these types of problems through the correct use of fuzzy logic.

Application of the FuGePSD method to the analysis of the salivary ^{1}H NMR dataset is performed with the same parameters used in the experimental study of Section 4 but using as local fitness the fuzzy confidence because the main goal of the experts is to obtain accurate subgroups and using three linguistic labels.

Results acquired served to segregate the total number of saliva specimens into five subgroups: four to describe active sore throat saliva specimens and one for saliva specimens corresponding to healthy patients can be observed in Table 9, where the representation of the subgroups and their values for the quality measures are presented. FuGePSD is able to obtain subgroups with a low number of variables to describe both values for the target variable highlighting the values in unusualness and trade-off sensitivity-confidence. Subgroups are precise in general with an average confidence equal to 85.78%. In addition, with these subgroups the support reached for the algorithm is very close to the total of examples (95%). The metabolic assignment for each ISB is shown in the foot-table.

Table 10
Sore throat disease subgroups detectable via application of the FuGePSD method.

Sb	ISB (ppm)	^{1}H NMR Resonance Mult.	Metabolic assignment	Sign of class. mean diff. (sore throat − control)	Statistical significance: ANOVA p value
1	6.31–6.33	m	Lipid oxidation product[a]	+	0.030
	0.60–0.62	Broad	Proteins	+	ns
	1.36–1.40	d	Acetoin-CH[3]	+	ns
	3.44–3.61	s	Glycine-α-CH_2	+	0.049
	6.83–6.88	Broad/d	Protein Tyrosine Residues – Unknown multiplet[a]	+	0.012
2	2.22–2.27	t	5-Aminovalerate-α-CH_2[b]	+	0.013
	2.29–2.31	Weak m	γ-Aminobutyrate-α-CH_2[a]/Propionylglycine-α-CH_2[a]	–	0.015
	2.78–2.83	m	Aspartate-β-CH_2	+	ns
	5.66–5.69	m	Senecioate-α-CH vinylic proton[a]	+	0.026
3	2.22–2.27	t	5-Aminovalerate-α-CH_2[b]	+	0.015
	2.78–2.83	m	Aspartate-β-CH_2	+	ns
	5.66–5.69	m	Senecioate-α-CH vinylic proton[a]	+	0.017
4	2.22–2.27	t	5-Aminovalerate-α-CH_2[b]	+	0.015
	5.66–5.69	m	Senecioate-α-CH vinylic proton[a]	+	0.017
	8.37–8.42	m	1-Methyladenine[a]/Pterin-pyrazine ring proton[a]	+	0.010
5	2.22–2.27	t	5-Aminovalerate-α-CH_2[b]	+	0.015
	5.66–5.69	m	Senecioate-α-CH vinylic proton[a]	+	0.017

Abbreviations: ISB, 'Intelligently-Selected' Bucket; s, singlet; d, doublet; t, triplet; m, multiplet; ns, not significant via mixed model ANOVA analysis; Mult, multiplicity.

[a] Tentative assignment (the 6.31–6.33 ppm ISB resonance may arise from a conjugated hydroperoxy- or hydroxydiene lipid oxidation product, and the 6.83–6.88 ppm multipet may arise from 3,4-dihydroxymandelate, 4-hydroxyphenylacetate, pyrocatechol or 3-hydroxymandelate);

[b] For a small number of samples, this ISB also contained an acetone-CH3 group signal (s, δ = 2.245 ppm).

Furthermore, for each subgroup an statistical analysis for each ISB is performed shown the *Metabolic Assignment*, the *Sign* of classification mean difference between both target values, and the *ANOVA pValue* in Table 10. Valuable biomarker features identified were proteins, including those with relatively intense tyrosine residue resonances, acetoin and glycine, whereas those for the four sore throat disease classifications included 5-aminovalerate and the amino acid L-aspartate. The identity of the 5-aminovalerate signals (i.e., those coupled to the intense $\delta = 2.24$ ppm one) were confirmed via the acquisition of both 1D and 2D COSY $^1H/^1H–^1H$ NMR profiles of the human salivary supernatant specimens. Indeed, the 2.24 ppm resonance was found to be clearly linked to those at $\delta = 1.66$ (two sets of overlapping *tt* multiplets) and 3.025 ppm (triplet) of relative intensities 2.0 and 1.0 respectively to that of the 2.24 ppm signal; these signals are ascribable to 5-aminovalerate's 3-/4- and 5-position methylene group protons, with the 2.24 ppm one assigned to the 1-position (α-CH_2) ones. With the exception of the 2.29–2.31 ppm spectral bucket, all of the ISBs selected as important disease-determining predictor variables were of a higher salivary concentration in the active sore throat disease class of patients than those in the healthy age-matched control group, and this may partially arise from dehydration, which is a common feature associated with this condition.

The clinical and metabolomic significance of the biomolecular features selected via application of the FuGePSD technique employed here will be reported and discussed in detail elsewhere. However, it should be noted that 5-aminovalerate, one of the key biomarkers detected, is a microbial metabolite generated by oral microflora via a mechanism involving the bacterial catabolism of L-lysine [24] (although it may also be formed endogenously [6]). Therefore, its elevated salivary concentration in patients with an acute sore-throat condition may reflect an enhanced (localised) level of microbial growth and preponderance in those afflicted. Moreover, acetoin was also found to be upregulated in the salivary metabolome of subjects with an acute sore-throat condition, and this agent is generated via fermentation processes; indeed, it is a catabolite of the butanediol cycle in microorganisms.

7. Concluding remarks

In this paper, a new proposal based on genetic programming for SD has been presented. The genetic programming together fuzzy logic bring a range of advantages to the FuGePSD algorithm:

- Flexibility in the generation of the individuals since they are constructed with tree structures and the variables are included in a dynamic manner. In this way, the use of genetic programming allows to evolve individuals without the necessity to include all variables in the representation facilitating the obtaining of descriptive rules.
- Compact rules set with a low number of variables are obtained through the use of different parameters in the algorithm.
- The use of the token competition operator contributes to the algorithm with diversity since it promotes the evolution of differing individuals, i.e. this operator forces individuals to seek their own niches in the search space extending the diversity.
- Facility the use with continuous variables due to the incorporation of fuzzy logic which is able to work with this type of variables without the necessity to make a previous discretisation.
- Finally, it is important to remark the optimisation of precision of the algorithm with the use of an operation scheme focused on an approach of cooperation and competition between the individuals of the population.

The main advantages of FuGePSD are shown in datasets with continuous variables in a wide experimental study supported by statistical tests. FuGePSD provides results which represent a marked improvement over those acquired with other SD algorithms. Specifically, the experimental study presented in this paper shows an improvement in SD task with respect to the best algorithm presented up to the moment NMEEFSD. Fuzzy subgroups obtained by FuGePSD are more precise and cover more examples of the problem. Subgroups isolated demonstrate improved relationships between sensitivity and confidence. Moreover, unusualness (which is a key quality measure in SD) is improved, with significant differences between to the others algorithms analysed. Finally, the contribution provides for this new proposal has been tested in a case study related with the acute sore throat obtaining very relevant results for researchers with expertise in this field.

Acknowledgment

This work was partially supported by the Spanish Ministry of Economy and Competitiveness under Projects TIN2012-33856 (FEDER Founds).

Profs. Grootveld, Elizondo and Carmona are very grateful to De Montfort University, Leicester, UK for the provision of a HEIF collaborative award to support this project.

References

[1] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, A. Verkamo, Fast discovery of association rules, in: U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy (Eds.), Advances in Knowledge Discovery and Data Mining, AAAI Press, 1996, pp. 307–328.
[2] A. Asuncion, D.J. Newman, UCI Machine Learning Repository, 2007. <http://www.ics.uci.edu/mlearn/MLRepository.html>.
[3] M. Atzmueller, F. Puppe, SD-Map – a fast algorithm for exhaustive subgroup discovery, in: Proceedings of the 17th European Conference on Machine Learning and 10th European Conference on Principles and Practice of Knowledge Discovery in Databases, LNCS, vol. 4213, Springer, 2006, pp. 6–17.

[4] M. Atzmueller, F. Puppe, H.P. Buscher, Towards knowledge-intensive subgroup discovery, in: Proceedings of the Lernen, Wissensentdeckung, Adaptivität, Fachgruppe Maschinelles Lernen, 2004, pp. 111–117.
[5] S. Bay, M. Pazzani, Detecting group differences: mining contrast sets, Data Min. Knowl. Disc. 5 (2001) 213–246.
[6] P.S. Callery, L.A. Geelhaar, Biosynthesis of 5-aminopentanoic acid and 2-piperidone from cadaverine and 1-piperideine in the mouse, J. Neurochem. 43 (6) (1984) 1631–1634.
[7] C.J. Carmona, C. Chrysostomou, H. Seker, M.J. del Jesus, Fuzzy rules for describing subgroups from Influenza A virus using a multi-objective evolutionary algorithm, Appl. Soft Comput. 13 (8) (2013) 3439–3448.
[8] C.J. Carmona, P. González, M.J. del Jesus, F. Herrera, NMEEF-SD: non-dominated multi-objective evolutionary algorithm for extracting fuzzy rules in subgroup discovery, IEEE Trans. Fuzzy Syst. 18 (5) (2010) 958–970.
[9] C.J. Carmona, P. González, M.J. del Jesus, F. Herrera, Overview on evolutionary subgroup discovery: analysis of the suitability and potential of the search performed by evolutionary algorithms, WIREs Data Min. Knowl. Disc. 4 (2) (2014) 87–103. http://dx.doi.org/10.1002/widm.1118.
[10] C.J. Carmona, P. González, M.J. del Jesus, C. Romero, S. Ventura, Evolutionary algorithms for subgroup discovery applied to e-learning data, in: Proceedings of the IEEE International Education Engineering, 2010, pp. 983–990.
[11] C.J. Carmona, P. González, B. García-Domingo, M.J. del Jesus, J. Aguilera, MEFES: an evolutionary proposal for the detection of exceptions in subgroup discovery. An application to concentrating photovoltaic technology, Knowl.-Based Syst. 54 (2013) 73–85.
[12] C.J. Carmona, S. Ramírez-Gallego, F. Torres, E. Bernal, M.J. del Jesus, S. García, Web usage mining to improve the design of an e-commerce website: OrOliveSur.com, Expert Syst. Appl. 39 (2012) 11243–11249.
[13] A. Claxson, M. Grootveld, C. Chander, J. Earl, P. Haycock, M. Mantle, S.R. Williams, C.J.L. Silwood, D.R. Blake, Examination of the metabolic status of rat air pouch inflammatory exudate by high field proton NMR spectroscopy, Biochim. Biophys. Acta-Molec. Basis Dis. 1454 (1) (1999) 57–70.
[14] K. Deb, A. Pratap, S. Agrawal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.
[15] M.J. del Jesus, P. González, F. Herrera, M. Mesonero, Evolutionary fuzzy rule induction process for subgroup discovery: a case study in marketing, IEEE Trans. Fuzzy Syst. 15 (4) (2007) 578–592.
[16] J. Demsar, Statistical comparisons of classifiers over multiple data sets, J. Learning Res. 7 (2006) 1–30.
[17] J. Derrac, C. Cornelis, S. García, F. Herrera, Enhancing evolutionary instance selection algorithms by means of fuzzy rough set based feature selection, Inform. Sci. 186 (1) (2012) 73–92.
[18] G.Z. Dong, J.Y. Li, Mining border descriptions of emerging patterns from dataset pairs, Knowl. Inform. Syst. 8 (2) (2005) 178–202.
[19] A.E. Eiben, J.E. Smith, Introduction to Evolutionary Computation, Springer, 2003.
[20] P. Espejo, S. Ventura, F. Herrera, A survey on the application of genetic programming to classification, IEEE Trans. Syst. Man Cybernet. – Part C: Appl. Rev. 40 (2) (2010) 121–144.
[21] A. Fernández, S. García, J. Luengo, E. Bernadó-Mansilla, F. Herrera, Genetics-based machine learning for rule induction: state of the art, taxonomy, and comparative study, IEEE Trans. Evol. Comput. 14 (6) (2010) 913–941.
[22] C. Ferri, J. Hernández-Orallo, R. Modroiu, An experimental comparison of performance measures for classification, Pattern Recogn. Lett. 30 (1) (2009) 27–38.
[23] D.B. Fogel, Evolutionary Computation – Toward a New Philosophy of Machine Intelligence, IEEE Press, 1995.
[24] J.C. Fothergill, J.R. Guest, Catabolism of l-lysine by *Pseudomonas aeruginosa*, J. Gen. Microbiol. 99 (1) (1977) 139–145.
[25] M. Friedman, The use of ranks to avoid the assumption of normality implicit in the analysis of variance, J. Am. Stat. Assoc. 32 (1937) 675–701.
[26] M.J. Gacto, M. Galende, R. Alcalá, F. Herrera, METSK-HDe: a multiobjective evolutionary algorithm to learn accurate TSK-fuzzy systems in high-dimensional and large-scale regression problems, Inform. Sci. 276 (2014) 63–79.
[27] D. Gamberger, N. Lavrac, Expert-guided subgroup discovery: methodology and application, J. Artif. Intell. Res. 17 (2002) 501–527.
[28] S. García, A. Fernández, J. Luengo, F. Herrera, Study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability, Soft Comput. 13 (10) (2009) 959–977.
[29] S. García, F. Herrera, An extension on "Statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons, J. Machine Learn. Res. 9 (2008) 2677–2694.
[30] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Longman Publishing Co., Inc., 1989.
[31] D.P. Greene, S.F. Smith, Competition-based induction of decision models from examples, Machine Learn. 13 (2–3) (1993) 229–257.
[32] M. Grootveld, M.D. Atherton, A.N. Sheerin, J. Hawkes, D.R. Blake, T.E. Richens, C.J.L. Silwook, E. Lynch, A.W.D. Claxson, In vivo absorption, metabolism, and urinary excretion of alpha,beta-unsaturated aldehydes in experimental animals. Relevance to the development of cardiovascular diseases by the dietary ingestion of thermally stressed polyunsaturate-rich culinary oils, J. Clin. Invest. 101 (6) (1998) 1210–1218.
[33] M. Grootveld, A. Sheerin, M. Atherton, A.D. Millar, E.J. Lynch, D.R. Blake, D.P. Naughton, Biomedical Applications of NMR Spectroscopy, vol. 25, John Wiley and Sons, 1996. Chapter: Applications of high resolution NMR analysis to the study of inflammatory diseases at the molecular level, pp. 295–327.
[34] F. Herrera, Genetic fuzzy systems: taxomony, current research trends and prospects, Evol. Intell. 1 (2008) 27–46.
[35] F. Herrera, C.J. Carmona, P. González, M.J. del Jesus, An overview on subgroup discovery: foundations and applications, Knowl. Inform. Syst. 29 (3) (2011) 495–525.
[36] J.H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, 1975.
[37] J.H. Holland, J.S. Reitman, Cognitive systems based on adaptive algorithms, in: D.A. Waterman, F. Hayes-Roth (Eds.), Pattern Directed Inference Systems, Academic Press, 1978, pp. 313–329.
[38] S. Holm, A simple sequentially rejective multiple test procedure, Scand. J. Stat. 6 (1979) 65–70.
[39] H. Ishibuchi, T. Nakashima, M. Nii, Classification and Modeling with Linguistic Information Granules: Advanced Approaches to Linguistic Data Mining, Springer, 2004.
[40] W. Kloesgen, Explora: a multipattern and multistrategy discovery assistant, in: Advances in Knowledge Discovery and Data Mining, American Association for Artificial Intelligence, 1996, pp. 249–271.
[41] J.R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press, 1992.
[42] P. Kralj-Novak, N. Lavrac, G.I. Webb, Supervised descriptive rule discovery: a unifying survey of constrast set, emerging pateern and subgroup mining, J. Machine Learn. Res. 10 (2009) 377–403.
[43] C.-S. Kuo, T.-P. Hong, C.-L. Chen, Applying genetic programming technique in classification trees, Soft Comput. 11 (12) (2007) 1165–1172.
[44] C.-S. Kuo, T.-P. Hong, C.-L. Chen, An improved knowledged-acquisition strategy based on genetic programming, Cybernet. Syst. 39 (7) (2008) 672–685.
[45] N. Lavrac, B. Cestnik, D. Gamberger, P.A. Flach, Decision support through subgroup discovery: three case studies and the lessons learned, Machine Learn. 57 (1–2) (2004) 115–143.
[46] N. Lavrac, P.A. Flach, B. Zupan, Rule evaluation measures: a unifying view, in: Proceedings of the 9th International Workshop on Inductive Logic Programming, LNCS, vol. 1634, Springer, 1999, pp. 174–185.
[47] A. Lemanska, M. Grootveld, C.J.L. Silwood, R.G. Brereton, Chemometric variance analysis of 1H NMR metabolomics data on the effects of oral rinse on saliva, Metabolomics 8 (1) (2011) 64–80.
[48] K.S. Leung, Y. Leung, L. So, K.F. Yam, Rule learning in expert systems using genetic algorithm: 1, concepts, in: K. Jizuka (Ed.), Proc. of the 2nd International Conference on Fuzzy Logic and Neural Networks, 1992, pp. 201–204.
[49] J.M. Luna, J.R. Romero, C. Romero, S. Ventura, On the use of genetic programming for mining comprehensible rules in subgroup discovery, IEEE Trans. Cybernet. 44 (12) (2014) 2329–2341.
[50] D. Martín, A. Rosete, J. Alcalá-Fdez, F. Herrera, QAR-CIP-NSGA-II: a new multi-objective evolutionary algorithm to mine quantitative association rules, Inform. Sci. 258 (2014) 1–28.

[51] B.L. Miller, D.E. Goldberg, Genetic algorithms, tournament selection, and the effects of noise, Complex Syst. 9 (1995) 193–212.
[52] R. Palm, H. Hellendoorn, D. Driankov, Model Based Fuzzy Control, Springer, 1997.
[53] W. Pedrycz, Fuzzy Modelling: Paradigms and Practices, Kluwer Academic Publishers, 1996.
[54] A.L. Bisno, M.A. Gerber, J.M. Gwaltney, E.L. Kaplan, R.H. Schwartz, Infectious Diseases Society of America, Practice guidelines for the diagnosis and management of group A streptococcal pharyngitis, Clin. Infect. Dis. 35 (2) (2002). pp. 113, 125.
[55] H.P. Schwefel, Evolution and Optimum Seeking, Sixth-Generation Computer Technology Series, Wiley, 1995.
[56] D. Sheskin, Handbook of Parametric and Nonparametric Statistical Procedures, second ed., Chapman and Hall/CRC, 2006.
[57] A. Siebes, Data surveying: foundations of an inductive query language, in: Proceedings of the 1st International Conference on Knowledge Discovery and Data Mining, AAAI Press, 1995, pp. 269–274.
[58] C.J.L. Silwood, E. Lynch, A.W.D. Claxson, M. Grootveld, 1H and 13C NMR spectroscopic analysis of human saliva, J. Dental Res. 81 (6) (2002) 422–427.
[59] S.F. Smith, A learning system based on genetic adaptive algorithms, Ph.D. thesis, Pittsburgh, PA, USA, 1980.
[60] G. Venturini, SIA: a supervised inductive algorithm with genetic search for learning attributes based concepts, in: Proceedings European Conference on Machine Learning, LNAI, vol. 667, Springer, 1993, pp. 280–296.
[61] C.H. Wang, T.P. Hong, S.S. Tseng, Integrating fuzzy knowledge by genetic algorithms, IEEE Trans. Evol. Comput. 2 (1998) 138–149.
[62] C.H. Wang, T.P. Hong, S.S. Tseng, C.M. Liao, Automatically integrating multiple rule sets in a distributed-knowledge environment, IEEE Trans. Syst. Man Cybernet. Part C 28 (3) (1998) 471–476.
[63] M.L. Wong, K.S. Leung, Data Mining using Grammar Based Genetic Programming and Applications, Kluwer Academics Publishers, 2000.
[64] K. Wongravee, G.R. Lloyd, C.J.L. Silwood, M. Grootveld, R.G. Brereton, Supervised self organizing maps (SOMs) for classification and variable selection: illustrated by application to NMR metabolomic profiling, Anal. Chem. 82 (2) (2010) 628–638.
[65] G. Worrall, There is a Lot of it About: Acute Respiratory Infection in Primary Care, Abingdon Engl: Radcliffe Publishing Ltd, 2006. Chapter: Acute Sore Throat, pp. 24–36.
[66] S. Wrobel, An algorithm for multi-relational discovery of subgroups, in: Proceedings of the 1st European Symposium on Principles of Data Mining and Knowledge Discovery, LNAI, vol. 1263, Springer, 1997, pp. 78–87.
[67] S. Wrobel, Inductive Logic Programming for Knowledge Discovery in Databases, Springer, 2001. Chapter: Relational Data Mining, pp. 74–101.
[68] L.A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning. Parts I, II, III, Inform. Sci. 8–9 (1975) 199–249. 301–357, 43–80.