Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 36 (2009) 1632–1644
Expert Systems with Applications
www.elsevier.com/locate/eswa

Evolutionary algorithms for subgroup discovery in e-learning: A practical application using Moodle data

C. Romero [a,*], P. González [b], S. Ventura [a], M.J. del Jesus [b], F. Herrera [c]

[a] *Department of Computer Sciences and Numerical Analysis, University of Córdoba, 14071 Córdoba, Spain*
[b] *Department of Computer Sciences, University of Jaén, 23071 Jaén, Spain*
[c] *Department of Computer Sciences and Artificial Intelligence, University of Granada, 18071 Granada, Spain*

Abstract

This work describes the application of subgroup discovery using evolutionary algorithms to the usage data of the Moodle course management system, a case study of the University of Cordoba, Spain. The objective is to obtain rules which describe relationships between the student's usage of the different activities and modules provided by this e-learning system and the final marks obtained in the courses. We use an evolutionary algorithm for the induction of fuzzy rules in canonical form and disjunctive normal form. The results obtained by different algorithms for subgroup discovery are compared, showing the suitability of the evolutionary subgroup discovery to this problem.

Keywords: Web-based education; Subgroup discovery; Evolutionary algorithms; Fuzzy rules

1. Introduction

The design and implementation of web-based education systems have grown exponentially in the last years, spurred by the fact that neither students nor teachers are bound to a specific location and that this form of computer-based education is virtually independent of any specific hardware platforms. These systems accumulate a great deal of information which is very valuable in analyzing students' behavior and assisting authors in the detection of possible errors, shortcomings and improvements. However, due to the vast quantities of data these systems can generate daily, it is very difficult to manage manually, and authors demand tools which assist them in this task, preferably on a continuous basis. The use of data mining is a promising area in the achievement of this objective (Romero & Ventura, 2006, 2007).

In the knowledge discovery in databases (KDD) process, the data mining step consists of the automatic extraction of implicit and interesting patterns from large data collections. A list of data mining techniques or tasks includes statistics, clustering, classification, outlier detection, association rule mining, sequential pattern mining, text mining, or subgroup discovery, among others (Klösgen & Zytkow, 2002).

In recent years, researchers have begun to investigate various data mining methods in order to help teachers improve e-learning systems. A review can be seen in (Romero & Ventura, 2007). These methods allow the discovery of new knowledge-based on students' usage data.

Subgroup discovery is a specific method for discovering descriptive rules (Klösgen, 1996; Wrobel, 1997). The objective is to discover characteristics of subgroups with respect to a specific property of interest (represented in the rule consequent). It must be noted that subgroup discovery aims at discovering individual rules (or local patterns of interest), which must be represented in explicit symbolic form and which must be relatively simple in order to be recognized as actionable by potential users. Therefore, the subgroups discovered in data have an explanatory nature and the interpretability for the final user of the

---

[*] Corresponding author. Fax: +34 957 218630.
E-mail address: cromero@uco.es (C. Romero).

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.11.026

extracted knowledge is a crucial aspect in this field. This task has been applied to different domains: detection of patient groups with risk for atherosclerotic coronary heart disease (Gamberger & Lavrac, 2002b), mining UK traffic data (Kavsek, Lavrac, & Bullas, 2002), personal web pages (Nakada & Kunifuji, 2003), identification of interesting diagnostic patterns to supplement a medical documentation and consultation system (Atzmueller, Puppe, & Buscher, 2004) or marketing problems (del Jesus, González, Herrera, & Mesonero, 2007).

This work proposes the application of subgroup discovery to the usage data of the course management system Moodle at the University of Cordoba, Spain. Moodle is a free open source course management system designed to help educators create effective online learning communities. Moodle has a flexible array of course activities such as forums, chats, quizzes, resources, choices, surveys, or assignments. Our objective is to obtain rules which describe relationships between the student's usage of the different activities and modules provided by this e-learning system and the final score obtained in the courses. These rules can help the teacher to discover beneficial or detrimental relationships between the use of web-based educational resources and the student's learning.

We will focus our attention in the use of a subgroup discovery algorithm-based on the use of genetic algorithms (GAs) called SDIGA (Subgroup Discovery Iterative Genetic Algorithm). SDIGA is an evolutionary model for the extraction of fuzzy rules for the subgroup discovery task. This algorithm is described in detail in (del Jesus et al., 2007). Its main characteristics are presented in this paper.

We compare the results obtained by this algorithm with those obtained by two classical subgroup discovery methods: Apriori-SD (Kavsek & Lavrac, 2006) and CN2-SD (Lavrac, Kavsec, Flach, & Todorovski, 2004). Furthermore, we also use an algorithm for class association rule discovery such as CBA (Classification Based on Association) (Liu, Hsu, & Ma, 1998). We will present an experimental study where SDIGA obtains the best results for our educational mining problem.

This paper is arranged in the following way: Section 2 describes the problem of discovering rules in e-learning and surveys some specific work in the area. Section 3 introduces the subgroup discovery task, the type of rules and quality measures used and the fuzzy evolutionary approach. Section 4 describes the e-learning case study, the experimentation carried out and the analysis of results. Finally, the conclusions and further research are outlined.

2. Rule discovery in learning management systems

Many web-based educational systems with different capabilities and approaches have been developed to deliver online education. There are different types of web-based educational systems: particular web-based courses, learning management systems, and adaptive and intelligent web-based educational systems (Romero & Ventura, 2006). This paper is mostly oriented forwards learning management systems. Different terms are used to denominate these systems: learning management systems (LMS), course management systems or learning content management systems. These systems are e-learning platforms that offer a great variety of channels and workspaces to facilitate information sharing and communication between participants in a course, allow educators to distribute information to students, produce content material, prepare assignments and tests, engage in discussions, manage distance classes and enable collaborative learning with forums, chats, file storage areas or news services. Some examples of commercial LMS are Blackboard, Virtual-U, WebCT, or TopClass among others and some examples of free LMS are Moodle, Ilias, Claroline, or ATutor. (Paulsen, 2003). These systems normally use a relational database to store the large data log of the students' activities and usage information. And although some platforms offer reporting tools, if there are a great number of students and a great amount of information, it becomes difficult for a tutor to extract useful information. Recently, some researchers propose using data mining techniques in order to help the tutor in this task.

Data mining techniques can be applied to analyzing student's usage data in order to identify useful patterns and to evaluate web activity to get more objective feedback for instruction and more knowledge about how the students learn on the LMS (Romero & Ventura, 2007). A data mining algorithm can discover knowledge using different representation models and techniques from two different perspectives:

- Predictive induction, whose objective is the discovery of knowledge for classification or prediction (Michie, Spiegelhalter, & Taylor, 1994). Classification rule discovery (Quinlan, 1993) or clustering (Han, Kamber, & Tung, 2001) are data mining tasks under the predictive induction approach.
- Descriptive induction, whose main objective is the extraction of interesting knowledge from data In this area, attention can be drawn to the discovery of association rules following an unsupervised learning model (Agrawal, Imielinski, & Swami, 1993), subgroup discovery (Klösgen, 1996; Wrobel, 1997) and other approaches to non-classificatory induction.

In the following, some of the most widely used data mining techniques in e-learning are described.

Classification is a supervised process of grouping physical or abstract objects into classes of similar characteristics. It belongs to predictive induction data mining methods. The objective of classification rules (Quinlan, 1993) is to obtain the necessary knowledge to create a classification system. The antecedents of these rules contain requirements (in the form of conditions), which match those objects that belong to the class label identified in the consequent of the rule. Classification rule mining has been applied to LMS in order to characterize the properties of a group of user profiles, similar pages or learning sessions, as can be seen in the following work.

Talavera and Gaudioso (2004) propose mining student's data using clustering in order to discover patterns reflecting user behaviors. They propose models for collaboration management to characterize similar behavior groups in unstructured collaboration spaces. Mor and Minguillon (2004) extend the sequencing capabilities of the SCORM standard to include the concept of recommended itinerary, by combining educators' expertise with learned experience acquired by system usage analysis. They use clustering algorithms for grouping students. Castro, Vellido, Nebot, and Minguillon (2005) detect atypical behavior in the grouping structure of the users of a virtual campus. They propose using a generative topographic mapping model and a clustering model to characterize groups of online students. The model neutralizes the negative impact of outliers on the data clustering process.

Association rule mining is one of the better-studied descriptive data mining methods. Its objective is to discover descriptive rules about relations between attributes of a set of data which overcome a user-specified confidence threshold (each rule must cover a minimum percentage of the data, that is the confidence threshold). Such rules associate one or more attributes of a dataset with another attribute, producing an if–then statement concerning attribute values. Mining association rules between sets of items in large databases was first proposed by Agrawal et al. (1993) and it opened up a brand new family of algorithms. The original problem was how to perform the market basket analysis which attempted to find all the interesting relationships between products bought in a given context. Association rule mining has been applied to LMS in order to reveal which contents students tend to access together, or which combination of tools they use.

We describe below some studies of the application of this technique to LMS. Wang (2002) develops a portfolio analysis tool-based on associative material clusters and sequences among them. This knowledge allows educators to study the dynamic browsing structure and to identify interesting or unexpected learning patterns. To do this, Wang discovers two types of relations: association relations and sequence relations between documents. Minaei-Bidgoli, Tan, and Punch (2004) propose mining interesting contrast rules for web-based education systems. Contrast rules help one to identify attributes characterizing patterns of performance disparity between various groups of students. Markellou, Mousourouli, Spiros, and Tsakalidis (2005) propose an ontology-based framework and discover association rules, using the Apriori algorithm. The role of ontology is to determine which learning materials are more suitable to be recommended to the user. Zaïane and Luo (2001) propose the discovery of useful patterns based on restrictions, to help educators evaluate students' activities in web courses. Li and Zaïane (2004) also use recommender agents for e-learning systems which use association rule mining to discover associations between user actions and URLs. The agent recommends online learning activities or shortcuts in a course web site based on a learner's access history. Lu (2004) uses fuzzy association rules in a personalized e-learning material recommender system. He uses fuzzy matching rules to discover associations between student's requirements and a list of learning materials. Romero, Ventura, and Bra (2004) propose using grammar-based genetic programming with multi-objective optimization techniques in order to provide a feedback to courseware authors. They discover interesting association rules by analyzing student's usage information. Merceron and Yacef (2004) use association rule and symbolic data analysis, as well as traditional SQL queries to mine student data captured from a web-based tutoring tool. Their goal is to find mistakes that often occur together. Freyberger, Heffernan, and Ruiz (2004) use association rules to guide a search for best fitting transfer models of student learning in intelligent tutoring systems. The association rules determine which operation to perform on the transfer model that predicts a student's success.

3. Subgroup discovery: classic approaches and evolutionary proposals

We have described some of the data mining techniques most used in e-learning, but subgroup discovery can also be applied to this task. In this section, the subgroup discovery task is introduced and classical and evolutionary approaches are described. First, we describe the topic of subgroup discovery and the classical approaches. Then, we analyze the use of evolutionary algorithms for rule induction. Finally, we introduce an evolutionary proposal for the subgroup discovery task.

3.1. Subgroup discovery

This section introduces the concept of subgroup discovery in the context of descriptive induction, describes classical approaches to the subgroup discovery task, and summarizes the quality measures typically used for this task.

3.1.1. Introduction to subgroup discovery

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

An induced subgroup description has the form of an implication,

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

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

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

The subgroup discovery task relies on the following main properties:

- The description language which specifies the subgroups must be appropriate in order to be applied effectively by the potential users. The subgroup description consists of a set of expressions. In the simplest case, each expression is single-valued; however negation or internal disjunctions are also possible.
- The quality function which measures the interest of the subgroup. A variety of quality functions have been proposed, as can be seen in (Klösgen, 1996), (Klösgen, 2002) and (Gamberger, Lavrac, & Krstacic, 2003). The applicable set of quality functions is determined by the type of target variable, the type of rule and the problem considered. In the next subsection we will describe several quality measures used in subgroup discovery algorithms.
- The search strategy. The search strategy is very important, since the dimension of the search space has an exponential relation to the number of features (or variables) and values considered.

3.1.2. Quality measures in subgroup discovery

A determining factor in the quality of any subgroup discovery algorithm is the quality measure to be used, both to select the rules and to evaluate the results of the process. Objective measures for descriptive induction evaluate each subgroup individually, but can be complemented by their variants in order to compute the mean of the induced set of descriptions of subgroups, thus allowing comparison between different subgroup discovery algorithms.

There have been different studies of objective quality measures for the descriptive induction process (Klösgen, 2002), (Piatetsky-Shapiro & Matheus, 1994), (Gamberger & Lavrac, 2002a) but it is difficult to reach an agreement on their use. Below, the more widely used quality measures in the specialized bibliography of subgroup discovery are described.

- *Coverage for a rule R^i* (Lavrac et al., 2004): measures the percentage of examples covered on average by one rule of the induced set of rules.

$$\text{Cov}(R^i) = \text{Cov}(\text{Cond}^i \rightarrow \text{Class}_j) = p(\text{Cond}^i) = \frac{n(\text{Cond}^i)}{n_s} \tag{1}$$

where $n(\text{Cond}^i)$ is the number of examples which verifies the condition *Cond^i* described in the antecedent (independently of the class to which belongs), and n_s is the number of examples.
The *average coverage for the set of rules* finally obtained is calculated using the following formula:

$$\text{COV} = \frac{1}{n_r}\sum_{i=1}^{n_r}\text{Cov}(R^i) \tag{2}$$

where n_r is the number of induced rules.
- *Support for a rule*: in descriptive induction processes the support for a rule is a standard measure which considers, by means of an expression that can vary in different proposals, the number of examples satisfying both the antecedent and the consequent parts of the rule. Lavrac et al. (2004) compute the overall support as the percentage of target examples (positive examples) covered by the rules. The support of a rule is therefore defined as the frequency of correctly classified examples covered.

$$\text{Sup}(\text{Cond}^i \rightarrow \text{Class}_j) = p(\text{Class}_j.\text{Cond}^i) = \frac{n(\text{Class}_j.\text{Cond}^i)}{n_s} \tag{3}$$

where $n(\text{Class}_j.\text{Cond}^i)$ is the number of examples which satisfy the conditions for the antecedent (Cond^i) and also belong to the value for the target variable (Class_j) indicated in the consequent part of the rule.
The *support* for a set of rules is computed by:

$$\text{SUP} = \frac{1}{n_s}\sum_{j=1}^{n_c} n(\text{Class}_j.\bigvee_{\text{Cond}^i \rightarrow \text{Class}_j}\text{Cond}^i) \tag{4}$$

where n_c is the number of values for the target variable considered. It must be noted that in this expression the examples which belong to various rules are considered only once.
- *Size (for a set of rules)*: the size of a set of rules is a complexity measure calculated as the number of induced rules (n_r). Complexity can also be measured as the mean

number of rules obtained for each class, or the mean of variables per rule.

- *Significance for a rule* (Klösgen, 1996): indicates the significance of a finding, if measured by the likelihood ratio of a rule.

$$\mathrm{Sig}(\mathrm{Cond}^i \to \mathrm{Class}_j) = 2 \cdot \sum_{j=1}^{n_c} n(\mathrm{Class}_j.\mathrm{Cond}^i) \cdot \log \frac{n(\mathrm{Class}_j.\mathrm{Cond}^i)}{n(\mathrm{Class}_j) \cdot p(\mathrm{Cond}^i)} \tag{5}$$

where $p(\mathrm{Cond}_i)$, computed as $n(\mathrm{Cond}_i)/n_s$, is used as a normalized factor.

It must be noted that, although each rule is for a specific class value, the significance measures the novelty in the distribution impartially, for all the class values.

The significance for a set of rules is computed as follows:

$$\mathrm{SIG} = \frac{1}{n_r} \sum_{i=1}^{n_r} \mathrm{Sig}(R^i) \tag{6}$$

- *Accuracy for a rule*: named "confidence" in descriptive data mining references, is the percentage of positive examples of a rule.

$$\mathrm{Acc}(\mathrm{Cond}^i \to \mathrm{Class}_j) = \frac{n(\mathrm{Class}_j.\mathrm{Cond}^i)}{n(\mathrm{Cond}^i)} \tag{7}$$

The accuracy for a set of rules is computed as:

$$\mathrm{ACC} = \frac{1}{n_r} \sum_{i=1}^{n_r} \mathrm{Acc}(R^i) \tag{8}$$

- *Unusualness for a rule*: is defined as the *weighted relative accuracy* of a rule (Lavrac, Flach, & Zupan, 1999).

$$\mathrm{WRAcc}(\mathrm{Cond}^i \to \mathrm{Class}_j) = \frac{n(\mathrm{Cond}^i)}{n_s} \cdot \left(\frac{n(\mathrm{Class}_j.\mathrm{Cond}^i)}{n(\mathrm{Cond}^i)} - \frac{n(\mathrm{Class}_j)}{n_s} \right) \tag{9}$$

The weighted relative accuracy of a rule can be described as the balance between the coverage of the rule ($p(\mathrm{Cond}^i)$) and its accuracy gain ($p(\mathrm{Class}_j.\mathrm{Cond}^i) - p(\mathrm{Class}_j)$).

The unusualness for a set of rules is computed as follows:

$$\mathrm{WRACC} = \frac{1}{n_r} \sum_{i=1}^{n_r} \mathrm{WRAcc}(R^i) \tag{10}$$

3.1.3. Related work in subgroup discovery

In the specialized bibliography, different methods have been developed which obtain descriptions of subgroups represented in different ways and using different quality measures. Here we briefly describe some of them:

- The first approach developed for subgroup discovery was EXPLORA (Klösgen, 1996). It uses decision trees for the extraction of rules. The rules are specified by defining a descriptive scheme and implementing a statistical verification method. The interest of the rules is measured using criteria such as evidence, generality, redundancy and simplicity.
- MIDOS (Wrobel, 1997) applies the EXPLORA approach to multirelational databases. It uses optimistic estimation and minimum support pruning. The goal is to discover subgroups of the target relation (defined as first order conjunctions) which have unusual statistical distributions with respect to the complete population. The quality measure is a combination of unusualness and size.
- SubgroupMiner (Klösgen, 2002) is an extension of EXPLORA and MIDOS. It is an advanced subgroup discovery system which uses decision rules and interactive search in the space of the solutions, allowing the use of very large databases by means of the efficient integration of databases, multirelational hypotheses, visualization based on interaction options, and the discovery of structures of causal subgroups. This algorithm uses as its standard quality function the classical binomial test to verify whether the statistical distribution of the target is significantly different in the extracted subgroup.
- SD (Gamberger & Lavrac, 2002a) is a rule induction system guided by expert knowledge: instead of defining an optimal measure to search for and select the subgroups automatically, the objective is to help the expert in performing flexible and effective searches on a wide range of optimal solutions.
- CN2-SD (Lavrac et al., 2004) (a modified version of the CN2 classification rule algorithm (Clark & Niblett, 1989)) induces subgroups in the form of rules using as quality measure the relation between true positives and false positives. CN2-SD uses a modified weighted relative accuracy as the quality measure for rule selection.
- RSD (Lavrac, Zelezny, & Flach, 2003), *Relational Subgroup Discovery*, has the objective of obtaining population subgroups which are as large as possible, with a statistical distribution as unusual as possible with respect to the property of interest, and which are different enough to cover most of the target population. It is a recent upgrade of the CN2-SD algorithm which enables relational subgroup discovery.
- APRIORI-SD (Kavsek & Lavrac, 2006) is developed by adapting to subgroup discovery the classification rule learning algorithm APRIORI-C (Jovanoski & Lavrac, 2001), a modification of the original APRIORI association rule learning algorithm (Agrawal et al., 1993). APRIORI-SD uses weighted relative accuracy as quality measure for the induced rules and probabilistic classification of the examples. For the evaluation of the set of rules the area under the *Receiver Operating Characteristic* (ROC) curve is used, in conjunction with the support and significance of each individual rule, and the size and accuracy of the set of rules.

- SD-Map (Atzmueller & Puppe, 2006) is an exhaustive subgroup discovery algorithm that uses the well-known FP-growth method (Han & Pei, 2000) for mining association rules with adaptations for the subgroup discovery task. SD-Map uses a modified FP-growth step that can compute the subgroup quality directly without referring to other intermediate results. The adaptations of the algorithms-based on Apriori for subgroup discovery are also valid for the FP-growth method.
- SDIGA (del Jesus et al., 2007) is an evolutionary fuzzy rule induction system which uses as quality measures for the subgroup discovery task adaptations of the measurements used in the association rules induction algorithms. Unlike all the other proposals, SDIGA uses linguistic rules as description language to specify the subgroups.

In this paper we will use the classical CN2-SD, APRIORI-SD and SDIGA to analyze and compare their results in the e-learning problem.

3.2. Evolutionary rule induction

Different proposals of Genetic Algorithms (GAs) have been developed for the extraction of rules of different types: classification, association or functional dependencies (Cordón, Herrera, Hoffmann, & Magdalena, 2001) (Ghosh & Jain, 2005).

The genetic representation of solutions is the most determinant aspect of any rule induction GA. In this sense, the proposals in the specialized literature follow two approaches in order to encode rules within a population of individuals:

- The "*Chromosome = Rule*" approach, in which each individual codifies a single rule.
- The "*Chromosome = Set of rules*", also called the Pittsburgh approach, in which each individual represents a set of rules. (Carse, Fogarty, & Munro, 1996) and (Wang, Hong, & Tseng, 1998) proposals use this representation model.

Within the "*Chromosome = Rule*" approach, three learning proposals can be found:

- The *Michigan* approach in which each individual codifies a single rule. Algorithms following this approach rule-based systems, which use a GA and a reinforcement component to learn rules that guide their performance in a certain environment (Kovacs, 2004).
- The *Iterative Rule Learning* (IRL) approach, in which each chromosome represents a rule, but the GA solution is the best individual obtained and the global solution is formed by the best individuals obtained when the algorithm is run multiple times. SLAVE (González & Pérez, 1999) and MOGUL (Cordón, del Jesus, Herrera, & Lozano, 1999) are proposals that follow this approach.
- The "*cooperative–competitive*" approach, in which the complete population or a subset of it codifies the rule-base. REGAL (Giordana & Neri, 1995) and LOGENPRO (Wong & Leung, 2002) use this type of representation.

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

3.3. SDIGA: an evolutionary algorithm for the induction of subgroup discovery rules

As we have mentioned, SDIGA is an evolutionary model for the extraction of fuzzy rules for the subgroup discovery task. This algorithm is described in detail in (del Jesus et al., 2007). Its main characteristics are presented below.

In the subgroup discovery task there is a set of descriptive variables and a single target variable describing the subgroups. As the objective is to obtain a set of rules describing subgroups for all the values of the target variable, the GA of this proposal discovers fuzzy rules with the consequent prefixed to one of the possible values of the target variable. In this way, each run of SDIGA obtains a set of rules corresponding to the value specified for the target variable, and the algorithm must be run for each one of the possible values of the target variable.

Each candidate solution is coded according to the "*Chromosome=Rule*" approach representing only the antecedent of the rule in the chromosome (since all the individuals of the population are associated with the same value of the target variable). The algorithm can be used to extract two types of rules: canonical or disjunctive normal form (DNF) rules. For the canonical rules, the antecedent of a rule is composed of a conjunction of variable-value pairs. On the other hand, a DNF rule offers a more flexible structure for the rules, allowing each variable to take more than one value, and facilitating the extraction of more general rules.

The core of SDIGA is a GA which uses a post-processing step based on a local search (a hill-climbing procedure). The hybrid GA extracts one simple and interpretable fuzzy rule. The post-processing step is applied in order to increase the generality of the extracted rule.

This hybrid GA is included in an iterative process for the extraction of a set of rules describing different parts (not necessarily apart) of the search space. A set of solutions generated in successive runs of the GA is obtained, corresponding with one value of the target variable.

Once the main features of the evolutionary algorithm have been outlined, the chromosome representation for the fuzzy rules, the fitness function and the hybrid model of the GA are described.

3.3.1. Fuzzy rules for descriptive data mining: chromosome representation

Interpretability is crucial in the field of data mining where knowledge should be extracted from data bases and represented in a comprehensible form, or for decision support systems where the reasoning process should be transparent to the user. In fact, the use of linguistic variables and linguistic terms in a discovery process has been explored by different authors (Ishibuchi, Nakashima, & Nii, 2004).

The fuzzy sets corresponding to the linguistic labels are defined by the corresponding membership functions which can be specified by the user or defined by a uniform partition if there is no expert knowledge available (using uniform partitions with triangular membership functions) (Zadeh, 1975).

To describe a fuzzy rule, we consider a subgroup discovery problem with:

- $\{X_m/m = 1,..,n_v\}$, a set of features used to describe the subgroups, where n_v is the number of features. These variables can be categorical or numerical;
- $\{\text{Class}_j/j = 1,\ldots,n_c\}$, a set of values for the target variable, where n_c is the number of values;
- $\{E^k = (e_1^k, e_2^k, \ldots, e_{n_v}^k, \text{class}_j)/k = 1,\ldots,n_s\}$, 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_s is the number of examples for the descriptive induction process;
- $X_m : \{LL_m^1, LL_m^2, \ldots, LL_m^{l_m}\}$, 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, R^i:Cond$^i \rightarrow$ Class$_j$, can be expressed as:

R^1 : If X_1 is LL_1^3 and X_7 is LL_7^1 then Class$_j$

where LL_7^1 is the linguistic label number 1 of the variable number 7.

We can also use fuzzy rules in disjunctive normal form (DNF) as description language to specify the subgroups. DNF rules permit a disjunction for the values of any variable present in the antecedent part. In this case, a DNF fuzzy rule can be expressed as:

R^1 : If X_1 is LL_1^1 or LL_1^3 and X_7 is LL_7^1 then Class$_j$

It must be noted that any subset of the complete set of variables (with any combination of linguistic labels related to the operator OR) can take part in the rule antecedent.

In this way a subgroup is a compact and interpretable description of patterns of interest in data.

One of the main aspects when working with fuzzy rules is the definition of membership functions associated with the fuzzy sets used. The fuzzy sets corresponding to the linguistic labels for a linguistic variable m, $(LL_m^1 \ldots LL_m^{l_m})$, are specified by means of the corresponding membership functions which can be defined by the user or defined by means of a uniform partition if the expert knowledge is not available. Fig. 1 shows a variable m with five linguistic labels using uniform partitions with triangular membership functions.

As we have mentioned previously, only the antecedent of the rule is represented in the chromosome and all the individuals are associated with the same value of the target variable.

For the canonical rules, in which the antecedent is composed of a conjunction of variable-value pairs, all the information relating to a rule can be contained in a fixed-length chromosome with an integer representation (the i-th position indicates the value adopted by the i-th variable). The set of possible values for the categorical features is increased with an additional value that indicates that the corresponding variable does not take part in the rule (value 0). For continuous variables the set of values is the set of linguistic terms determined heuristically or with expert information, plus the value indicating the absence of the variable. Fig. 2 shows a chromosome which follows this representation for the antecedent of the rule:

If X_1 is LL_1^3 and X_3 is LL_3^1 then Class$_2$

For the DNF rules, in which each variable can take more than one value, the antecedent of a rule can be represented by a fixed-length chromosome with a binary representation in which, for each feature a bit for each of the possible values of the feature is stored; in this way, if the corresponding bit contains the value 0 it indicates that the value is not used in the rule, and if the bit contains the value 1 it indicates that the corresponding value is included. Fig. 3 shows a chromosome using this representation for the antecedent of the rule:

If X_1 is LL_1^1 or LL_1^3 and X_3 is LL_3^1 then Class$_2$

3.3.2. GA fitness function

The fitness function of the GA combines, according to the following expression, three factors: accuracy, coverage and significance of the rule:

$$\text{fitness}(c) = \frac{\omega_1 \cdot \text{Accu}(c) + \omega_2 \cdot \text{Cov}(c) + \omega_3 \cdot \text{Sign}(c)}{\omega_1 + \omega_2 + \omega_3}$$

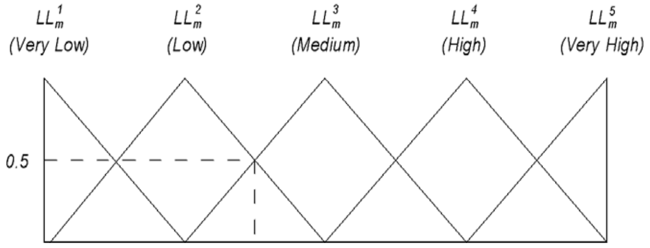

Fig. 1. Example of fuzzy partition for a continuous variable.

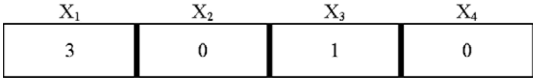

Fig. 2. Encoding model of a canonical rule.

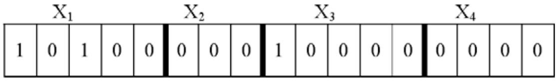

Fig. 3. Encoding model of a DNF rule.

These measures are computed in the following way:

- *Accuracy (Accu)*: the percentage of positive examples, i.e., the confidence as was defined in (7).
- *Coverage (Cov)*: the percentage of examples (both positive and negative) covered by the rule, as defined in (1).
- *Significance (Sign)*: indicates the significance of a finding, if measured by the likelihood ratio of a rule, as defined in (5).

The overall objective of the evaluation function is to direct the search towards rules which maximize accuracy and significance, minimizing the number of negative and non-covered examples.

3.3.3. GA hybrid model

The GA uses a modified steady-state reproduction model, with the aim of increasing the diversity of the population. In this model, the original population is modified through the substitution of the worst individuals by individuals resulting from crossover and mutation. Recombination is achieved by means of a two-point crossover operator and a biased random mutation operator.

Crossover is applied to the two best individuals of the population, obtaining two new individuals, who will substitute the two worst individuals in the population.

Mutation is carried out by means of a biased random mutation operator applied to the gene selected according to the mutation probability. This operator can be applied in two different ways: in the first the mutation causes the elimination of the variable to which the gene corresponds, and in the second the value for the variable to be muted is randomly assigned. The selection of one of the two choices is performed randomly, with the same probability. The use of this operator allows the promotion of diversity in the population.

Finally, a post-processing step is applied to the obtained rule, improving it through a hill-climbing process, modifying the rule in order to increase the degree of support. In order to accomplish this, in each iteration a variable is selected so that when it is eliminated, the support of the resulting rule is increased, thus obtaining more general rules. The optimized rule will substitute the original one only if it overcomes minimum confidence.

4. E-learning case study: usage data of the cordoba university moodle e-learning system

In this section we examine the Moodle case study. We first describe our specific problem and then show the experimental results obtained in the execution of the different subgroup discovery algorithms. Finally we analyze several rules from the point of view of the teacher with the aim of improving the e-learning courses.

4.1. Problem description

As we have mentioned previously, we have used the students' usage data of the Moodle system, which is one of the most widely used e-learning systems (Flate, 2003). Moodle has a large and diverse user's community with over 75 languages in over 160 countries (Moodle, 2007).

Our main objective in using subgroup discovery in an e-learning system is to analyze what relation the usage of complementary activities of a course can have to final mark obtained. We have used the final mark as the variable to characterize.

We are going to use different subgroup discovery algorithms in order to evaluate the results obtained and to analyze which algorithm discovers the information of highest interest/usability to the teacher of the course. Our objective is to present the results to the teacher in the form of rules in order to allow the use of this knowledge in the decision making concerning the complementary activities of the course. For example the teacher can decide to promote the use of some type of activities to obtain a high mark, or on the contrary eliminate some activities because they are associated with low marks.

The Moodle system contains a great deal of detailed information on course content, users, usage, etc., stored in a relational data base. We have applied a pre-processing step to the information, obtaining a new summary table (see Table 1) with the most important information related to our objective.

Table 1 contains a summary of the activities completed and the mark obtained by each student in an e-learning course. We have discretized the marks into classes (fail, pass, good and excellent) in order to codify them as the values of the rule consequent.

We have access to information corresponding to 192 different courses of the University of Cordoba. Among all these courses, we have however chosen only the 5 courses (with a total of 293 students) with the highest usage of the activities and resources available in Moodle.

We implemented the subgroup discovery algorithm for the KEEL data mining platform, available via the web (KEEL, 2007). We then exported all the information of the summary table to a text file using the same forma as the KEEL platform (Alcalá et al., 2007). This format is similar to the Weka ARFF format (Witten & Frank, 2005) and has two different sections: the first section contains the header information (the name of the relation, a

Table 1
Attributes used for each student

Name	Description
Course	Identification of the course
*n*_assignment	Number of assignments completed
*n*_assignment_a	Number of assignments passed
*n*_assignment_s	Number of assignments failed
*n*_quiz	Number of quizzes completed
*n*_quiz_a	Number of quizzes passed
*n*_quiz_s	Number of quizzes failed
*n*_messages	Number of messages sent to the chat
*n*_messages_ap	Number of messages sent to the teacher
*n*_posts	Number of messages sent to the forum
*n*_read	Number of forum messages read
Mark	Discretized student's mark

list of the attributes and their types); the second contains the data information (the data declaration line and the actual instance lines, a row for each student in which there is a column for each attribute value). An example of the KEEL format in which we present our summary dataset table is shown in Table 2.

4.2. Experimental results of the application of subgroup discovery algorithms

In order to verify the applicability of our proposal, we have compared the results of the SDIGA algorithm with those of other subgroup discovery algorithms:

- The classical subgroup discovery algorithms Apriori-SD (Kavsek & Lavrac, 2006) and CN2-SD (Lavrac et al., 2004).
- CBA algorithm (Liu et al., 1998), that discovers a special type of association rules (with a single predetermined target) called class association rules (CARs).

CBA, Apriori-SD and CN2-SD are deterministic algorithms, whereas SDIGA is nondeterministic. In order to carry out the comparison, we have followed the following procedure: for the classic deterministic algorithms we have performed a set of runs, varying one of their parameters each time. In the case of CBA and Apriori-SD, we have used 4 minimum confidence values (0.6, 0.7, 0.8 and 0.9) with a minimum support of 0.2. In the case of the CN2-SD, we have used the γ parameter (0.9, 0.7, 0.5 and additive) with a star size of 2.

Table 2
Summary dataset table in KEEL format

```
@relation student_summarization
@attribute course {C29, C46, C88, C110, C111}
@attribute n_assignment integer
...
@attribute mark {FAIL, PASS, GOOD, EXCELLENT}
@data
C110,10,10,6,0,12,9,3,0, GOOD
C110,9,9,0,0,11,8,3,0, PASS
C110,11,11,0,0,13,5,8,0, FAIL
C110,11,11,0,0,11,6,5,0, FAIL
C110,13,13,7,0,0,0,0,0, EXCELLENT
C110,8,8,7,0,7,6,1,0, FAIL
```

For the nondeterministic SDIGAs we have performed 5 different runs for a set of values of minimum confidence (0.6, 0.7, 0.8 and 0.9) using the following parameters:

- Population size: 100.
- Maximum number of evaluations of individuals in each GA run: 10,000.
- Crossover probability: 0.6.
- Mutation probability: 0.01.
- Number of linguistic labels for the continuous variables: 5 (very high, high, medium, low, very low).
- Weights for the fitness function: 3 for accuracy, 1 for coverage and 4 for significance. This set of weights has been chosen according to the results obtained in an experimental study.

As we have mentioned before, SDIGA can be used to obtain two types of rules, DNF and non-DNF (or canonical) rules. We have carried out the experiments for both types of rules.

Table 3 shows the results obtained by the classic algorithms with their different parameter values and the averages of the 5 runs of the SDIGAs (both types of rules, noted as SDIGA non-DNF and SDIGA DNF) with each value of minimum confidence. The table shows the total number of rules obtained, the number of attributes in the antecedents of the rules and the values of the quality measures. These quality measures are:

- Coverage of the rule set, COV, as defined in (2).
- Significance of the rule set, SIG, as defined in (6).
- Accuracy of the rule set, ACC, as defined in (8).

Analyzing the number of rules and attributes in Table 4 we can observe that:

- SDIGA non-DNF, SDIGA DNF and Apriori-SD algorithms discover the lowest number of rules (with very similar values); on the other hand CN2-SD and more specially CBA discover the highest number of rules.
- Regarding the number of attributes, Apriori-SD, CBA and SDIGA non-DNF obtain a low number of attributes (with very similar values) followed by SDIGA DNF. Finally, CN2-SD obtains the highest number of attributes.

From our problem's point of view we are interested in discovering a low number of rules with few attributes in order to facilitate the comprehensibility of these rules to the teacher. So the CBA and CN2-SD algorithms are not the most appropriate for our problem due to the fact that one discovers too many rules and the other discovers rules with too many attributes.

Table 3
Experimental results of the algorithms

Algorithm	Number of rules	Number of Attributes	Coverage	Significance	Accuracy
CBA CfMin 0.6	205	1.8	0.3536	25.5559	0.6130
CBA CfMin 0.7	201	1.9	0.3665	28.9316	0.6130
CBA CfMin 0.8	142	2.2	0.2568	42.1091	0.6952
CBA CfMin 0.9	60	2.5	0.0094	35.8586	0.6778
Apriori-SD CfMin 0.6	8	1.0	0.6220	26.1321	0.6130
Apriori-SD CfMin 0.7	9	1.3	0.6685	29.5409	0.6130
Apriori-SD CfMin 0.8	6	1.5	0.3613	42.1091	0.6130
Apriori-SD CfMin 0.9	5	2.0	0.2253	36.8100	0.6312
CN2-SD ($\gamma = 0.5$)	13	5.5	0.4151	44.9486	0.7157
CN2-SD ($\gamma = 0.7$)	17	5.5	0.3980	48.4380	0.7191
CN2-SD ($\gamma = 0.9$)	16	5.3	0.3878	50.2812	0.7294
CN2-SD (add)	32	5.7	0.5084	54.4237	0.7123
SDIGA non-DNF CfMin 0.6	7.8	2.0	0.0878	21.9924	0.8088
SDIGA non-DNF CfMin 0.7	6.2	2.1	0.0766	16.7925	0.7502
SDIGA non-DNF CfMin 0.8	6.0	2.2	0.1271	25.2460	0.7792
SDIGA non-DNF CfMin 0.9	4.8	2.0	0.1288	33.8351	0.7546
SDIGA DNF CfMin 0.6	7.8	3.1	0.3071	40.6689	0.7575
SDIGA DNF CfMin 0.7	9.4	3.6	0.2629	45.5539	0.8162
SDIGA DNF CfMin 0.8	7.4	3.2	0.2625	42.9744	0.7598
SDIGA DNF CfMin 0.9	5.4	3.0	0.1639	25.7814	0.7882

Analyzing the quality measures in Table 3 we can observe that:

- Accuracy (or confidence) measure indicates in our case the number of students that fulfill the rule antecedent and correspond to the associated class (the number of examples associated with the class). SDIGA DNF and non-DNF show the highest values followed by CN2-SD, CBA and Apriori-SD.
- Coverage is, like support, a measure of the generality of the rule. In our case it measures the number of students that fulfill the rule antecedent. Apriori-SD and CN2-Cd show the highest values, followed by SDIGA DNF, CBA and SDIGA non-DNF.
- Significance is a measure of the quantitative relevance and the interest of the rule. CN2-SD and SDIGA DNF show the highest values, followed by Apriori-SD, CBA, and SDIGA non-DNF.

The most desirable algorithm with regard to the values of these three quality measures would be an algorithm that simultaneously shows the highest values for all the measures. As we have seen there is not a single algorithm which achieves this. Among these measures, accuracy is arguably the most important, as it represents the reliability or confidence of the rule. In order to make reliable decisions the teacher has to use high confidence rules. We can observe in Table 3 that both versions of SDIGA discover rules with high accuracy and reasonable significance.

4.3. Using the rules obtained by SDIGA

In this section, we will describe some examples of the rules obtained and how these can be useful in course decision-making. The subgroup discovery rules reveal interesting information on the behavior of the students, and can help the teacher to discover beneficial or detrimental relationships between the use of web-based educational resources and the student's learning. The instructor can use the knowledge discovered by these rules to make decisions about Moodle course activities.

First, we describe several examples of rules discovered by the SDIGA non-DNF algorithm and analyze their meanings from a pedagogical point of view with the aim of improving the course.

IF course = C110 AND n_assignment = High
AND n_posts = High
THEN mark = Good
(Accuracy: 0.9285, Significance: 6.5348, Coverage: 0.1575)

This rule shows that in the *ProjectManagement* (C110) course, the students who have completed a high number of assignments and sent a lot of messages to the forum, have also obtained good marks. The teacher must continue to promote these types of activity in this course because of their effectiveness for the students in the final mark obtained.

IF course = C29 AND n_messages_ap = Very low
THEN mark = Fail
(Accuracy: 0.8560, Significance: 59.1774, Coverage: 0.2520)

In the *AppliedComputerScienceBasis* (C29) course, most of the students who have sent a very low number of messages to the teacher have failed. Using this information, the teacher can direct more attention to these students because they have a higher probability of failing.

It is important to point out that we have also discovered rules that provide information which is fairly obvious to the teacher. Some examples of this type of rules are the following:

IF n_quiz_a = Very low
THEN mark = Fail
(Accuracy: 0.6280, Significance: 6.500, Coverage: 0.0205)

This rule shows that if the number of quizzes passed is very low then the final mark obtained is fail. This rule is completely logical for the teacher (students who do not pass on-line quizzes are unlikely to pass a pen and pencil exam), and it does not provide any new information on how to improve the course.

IF n_quiz_a = Very high
THEN mark = Excellent
(Accuracy: 0.7819, Significance: 35.7308, Coverage: 0.1280)

This is the opposite rule to the previous one and states that the students that obtain a very high score in the quizzes then they logically obtain excellent marks.

Below we describe some examples of rules discovered by the SDIGA DNF algorithm.

IF course = C110 OR C88
AND n_posts = High OR Very High
AND n_quiz_a = Medium OR High OR Very High
THEN mark = Good
(Accuracy: 0.7382, Significance: 43.4771, Coverage: 0.2431)

This rule shows that if the students of the course *ProjectManagement* (C110) or *ComputerScienceBasis* (C88) have sent a high or very high number of messages to the forum, and they have also obtained a medium, high or very high score in the quizzes, then they obtain good marks.

IF course = C29 OR C110 OR C111
AND n_assignment_s = Very High OR High OR Medium
AND n_quiz_s = Very High OR High OR Medium
AND n_messages_ap = Very low OR Low
THEN mark = Fail
(Accuracy: 0.8667, Significance: 61.8034, Coverage: 0.4726)

This rule shows that if the students of the course *ProgrammingForEngineers* (C29) or *ProjectManagement* (C110) or *ComputerScienceBasis* (C88) have failed in a very high, high or medium number of assignments, have failed in a very high, high or medium number of quizzes, and have sent a very low or low number of messages to the teacher, then they have obtained a fail in their final marks.

These rules belonging to the set of rules discovered by SDIGA DNF present a higher coverage value than the previous ones. They use more than one value per variable in the rule, allowing us to cover more examples.

5. Conclusions

In this work we have described the application of subgroup discovery to e-learning, with the case study of the Moodle course management system. We have used real usage data pickep up from students at the University of Cordoba, Spain.

We have compared the results obtained by different algorithms for subgroup discovery, showing the suitability of the evolutionary subgroup discovery to this problem. In particular, SDIGA algorithm obtains a small number of rules which are highly understandable for the teacher. It also obtains similar results in the rules' quality measures and optimum results in the accuracy of the rules.

Our final objective is to show the discovered rules and theirs measures to the teacher, so that he can decide on course improvement. We have shown how the teacher can make decisions concerning the courses' activities and type of students in order to improve the course using the information provided by these rules.

Acknowledgement

The authors gratefully acknowledge the financial support provided by the Spanish department of Research under TIN2005-08386-C05-01, TIN2005-08386-C05-02 and TIN2005-08386-C05-03 Projects.

References

Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on management of data* (pp. 207–216). Washington, DC.

Alcalá, J., Sánchez, L., García, S., del Jesus, M. J., Ventura, S., Garrell, J. M., et al. (2007). KEEL: A data mining software tool for assessing the performance of knowledge extraction-based on evolutionary algorithms. *Soft computing, to appear.*

Atzmueller, M., Puppe, F. & Buscher, H. P. (2004). Towards knowledge-intensive subgroup discovery. In *Proceedings Lernen, Wissensentdeckung und Adaptivität Workshop (LWA'04)* (pp. 117–123). Berlin, Germany.

Atzmueller, M., & Puppe, F. (2006). SD-Map – A fast algorithm for exhaustive subgroup discovery. In *Proceedings of the 10th European conference on principles and practice of knowledge discovery in databases (PKDD 2006). Lecture notes in computer science* (Vol. 4213, pp. 6–17). Berlin: Springer-Verlag.

Carse, B., Fogarty, T. C., & Munro, A. (1996). Evolving fuzzy rule based controllers using genetic algorithms. *Fuzzy Sets and Systems, 80*, 273–293.

Castro, F., Vellido, A., Nebot, A., & Minguillon, J. (2005). Detecting atypical student behaviour on an e-learning system. In *I Simposio Nacional de Tecnologas de la Información y las comunicaciones en la Educacion* (pp. 153–160). Granada, Spain.

Clark, P., & Niblett, T. (1989). The CN2 induction algorithm. *Machine Learning, 3*(4), 261–283.

Cordón, O., del Jesus, M. J., Herrera, F., & Lozano, M. (1999). MOGUL: A methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning approach. *International Journal of Intelligent Systems, 14*, 1123–1153.

Cordón, O., Herrera, F., Hoffmann, F., & Magdalena, L. (2001). Genetic fuzzy systems: evolutionary tuning and learning of fuzzy knowledge bases. World Scientific.

del Jesus, M. J., González, P., Herrera, F., & Mesonero, M. (2007). Evolutionary fuzzy rule induction process for subgroup discovery: a case study in marketing. *IEEE Transactions on Fuzzy Systems, 15*(4), 578–592.

Flate, M. (2003). *Online education and learning management systems. Global e-learning in a Scandinavian perspective*. Oslo: NKI Forlaget.

Freyberger, J., Heffernan, N., & Ruiz, C. (2004). Using association rules to guide a search for best fitting transfer models of student learning. In *Workshop on analyzing student–tutor interactions logs to improve educational outcomes at ITS conference*. Maceio (pp. 1–4). Brazil.

Gamberger, D. & Lavrac, N. (2002). Descriptive induction through subgroup discovery: A case study in a medical domain. In *Proceedings of the 19th international conference on machine learning (ICML 2002)* (pp. 163–170). Sydney, Australia.

Gamberger, D., & Lavrac, N. (2002a). Expert-guided subgroup discovery: Methodology and application. *Journal Of Artificial Intelligence Research, 17*, 1–27.

Gamberger, D., Lavrac, N., & Krstacic, G. (2003). Active subgroup mining: A case study in coronary heart disease risk group detection. *Artificial Intelligence in Medicine, 28*(1), 27–57.

Ghosh, A., & Jain, L. (2005). *Evolutionary computation in data mining. Studies in fuzziness and soft computing* (Vol. 163). Springer.

Giordana, A., & Neri, F. (1995). Search-intensive concept induction. *Evolutionary Computation, 3*(4), 375–416.

González, A., & Pérez, R. (1999). SLAVE: A genetic learning system based on an iterative approach. *IEEE Transactions on Fuzzy Systems, 7*(2), 176–191.

Han, J., Kamber, M., & Tung, A. K. H. (2001). *Spatial clustering methods in data mining: A survey*. Philadelphia: Taylor & Francis.

Han, J., & Pei, J. (2000). Mining frequent patterns by pattern-growth: Methodology and implications. *ACM SIGKDD Explorations Newsletter, 2*(2), 14–20.

Ishibuchi, H., Nakashima, T., & Nii, M. (2004). *Classification and modeling with linguistic information granules*. Springer-Verlag.

Jovanoski, V., & Lavrac, N. (2001). Classification rule learning with APRIORI-C. In *progress in artificial intelligence knowledge extraction, Multi-agent systems, Logic programming, and Constraint solving: 10th Portuguese Conference on Artificial Intelligence (EPIA'01)* (pp. 44–51). Porto, Portugal.

Kavsek, B., Lavrac, N., & Bullas, J. C. (2002). Rule induction for subgroup discovery: a case study in mining UK traffic accident data. In *International Multi-Conference on Information Society (IS'02)* (pp. 127–130). Ljubljana, Slovenia.

Kavsek, B., & Lavrac, N. (2006). APRIORI-SD: Adapting association rule learning to subgroup discovery. *Applied Artificial Intelligence, 20*(7), 543–583.

KEEL (2007). Available from <http://www.keel.es>.

Klösgen, W. (1996). Explora: A multipattern and multistrategy discovery assistant. In U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Eds.), *Advances in knowledge discovery and data mining* (pp. 249–271). Menlo Park, California: AAAI Press.

Klösgen, W. (2002). Subgroup discovery. In W. Klösgen & J. Zytkow (Eds.), *Handbook of data mining and knowledge discovery* (pp. 354–364). NY: Oxford University Press.

Klösgen, W., & Zytkow, J. (2002). *Handbook of data mining and knowledge discovery*. NY: Oxford University Press.

Kovacs, T. (2004). *Strength or accuracy: Credit assignment in learning classifier systems*. Springer-Verlag.

Lavrac, N., Flach, P., & Zupan, B. (1999). Rule evaluation measures: A unifying view. In *Proceedings of the 9th international workshop on inductive logic programming (ILP'99)* (pp. 174–185).

Lavrac, N., Kavsec, B., Flach, P., & Todorovski, L. (2004). Subgroup discovery with CN2-SD. *Journal of Machine Learning Research, 5*, 153–188.

Lavrac, N., Zelezny, F., & Flach, P. A. (2003). RSD: Relational subgroup discovery through first-order feature construction. In *Proceedings of the international conference on inductive logic programming. Lecture notes on artificial intelligence* (2583, pp. 149–165). Sydney, Australia: Springer.

Liu, B., Hsu, W., & Ma, Y. (1998). Integrating classification and association rule mining. *In Proceedings of the international conference on knowledge discovery and data ming* (pp. 80–86). NY.

Li, J., & Zaïane, O. (2004). Combining usage, content, and structure data to improve web site recommendation. In *Proceedings of the international conference on e-commerce and web technologies. Lecture notes on computer science* (3182, pp. 305–315). Zaragoza, Spain: Springer.

Lu, J. (2004). Personalized e-learning material recommender system. In *Proceedings of the international conference on information technology for application* (pp. 374–379). London, England.

Markellou, P., Mousourouli, I., Spiros, S., & Tsakalidis, A. (2005). Using semantic web mining technologies for personalized e-learning experiences. In *Proceedings of the web-based education* (pp. 461–826). Grindelwald, Switzerland.

Merceron, A., & Yacef, K. (2004). Mining student data captured from a web-based tutoring tool: Initial exploration and results. *Journal of Interactive Learning Research, 15*(4), 319–346.

Michie, D., Spiegelhalter, D. J., & Taylor, C. C. (1994). *Machine learning, neural and statistical classification*. Ellis Horwood.

Minaei-Bidgoli, B., Tan, P., & Punch, W. (2004). Mining interesting contrast rules for a web-based educational system. In *Proceedings of the international conference on machine learning applications* (pp. 1–8). Louisville, KY.

Moodle (2007). Available from <http://moodle.org>.

Mor, E., & Minguillon, J. (2004). E-learning personalization based on itineraries and long-term navigational behavior. In *Proceedings of the 13th international world wide web conference* (pp. 264–265). NY.

Nakada, T., & Kunifuji, S. (2003). Subgroup discovery among personal homepages. In *Proceedings of the 6th international conference on discovery science (DS 2003)* (pp. 385–392). Sapporo, Japan.

Paulsen, M. (2003). *Online education and learning management systems*. Bekkestua: NKI Forlaget.

Piatetsky-Shapiro, G., & Matheus, C. (1994). The interestingness of deviation. In *Proceedings of the AAAI workshop on knowledge discovery in databases* (pp. 25–36). Seattle, Washington.

Quinlan, J. R. (1993). *C4.5: Programs for machine learning*. San Mateo, CA: Morgan Kaufmann.

Romero, C., & Ventura, S. (2006). *Data mining in e-learning*. Southampton, UK: Wit Press.

Romero, C., & Ventura, S. (2007). Educational data mining: A survey from 1995 to 2005. *Expert Systems with Applications, 33*(1), 135–146.

Romero, C., Ventura, S., & Bra, P. D. (2004). Knowledge discovery with genetic programming for providing feedback to courseware author. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research, 14*(5), 425–464.

Talavera, L., & Gaudioso, E. (2004). Mining student data to characterize similar behavior groups in unstructured collaboration spaces. In *Workshop on artificial intelligence in CSCL. 16th European conference on artificial intelligence* (pp. 17–23). Valencia, Spain.

Wang, F. (2002). On using data-mining technology for browsing log file analysis in asynchronous learning environment. In *Proceedings of the Conference on educational multimedia, hypermedia and telecommunication* (pp. 2005–2006). Denver, Colorado.

Wang, C. H., Hong, T. P., & Tseng, S. S. (1998). Integrating fuzzy knowledge by genetic algorithms. *IEEE Transactions on Evolutionary Computation, 2*, 138–149.

Witten, I. H., & Frank, E. (2005). *Data mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Wong, M. L., & Leung, K. S. (2002). *Data mining using grammar based genetic programming and applications*. Kluwer Academics Publishers.

Wrobel, S. (1997). An algorithm for multi-relational discovery of subgroups. In *Proceedings of conference principles of data mining and knowledge discovery* (pp. 78–87). London, UK.

Zadeh, L. A. (1975). The concept of a linguistic variable and its applications to approximate reasoning, Parts I, II, III. *Information Sciences, 8–9*, pp. 199–249, 301–357, 43–80.

Zaïane, O., & Luo, J. (2001). Web usage mining for a better web-based learning environment. In *Proceedings of conference on advanced technology for education* (pp. 60–64). Banff, Alberta.