Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 35 (2008) 1949–1965
Expert Systems with Applications
www.elsevier.com/locate/eswa

Making CN2-SD subgroup discovery algorithm scalable to large size data sets using instance selection ☆

José-Ramón Cano [a], Francisco Herrera [b], Manuel Lozano [b], Salvador García [b,*]

[a] Department of Computer Science, University of Jaén, 23700 Linares, Jaén, Spain
[b] Department of Computer Science and Artificial Intelligence, University of Granada, 18071 Granada, Spain

Abstract

The subgroup discovery, domain of application of CN2-SD, is defined as: "given a population of individuals and a property of those individuals, we are interested in finding a population of subgroups as large as possible and have the most unusual statistical characteristic with respect to the property of interest".

The subgroup discovery algorithm CN2-SD, based on a separate and conquer strategy, has to face the scaling problem which appears in the evaluation of large size data sets. To avoid this problem, in this paper we propose the use of instance selection algorithms for scaling down the data sets before the subgroup discovery task. The results show that CN2-SD can be executed on large data set sizes pre-processed, maintaining and improving the quality of the subgroups discovered.

Keywords: Subgroup discovery; Scaling down; Instance selection

1. Introduction

In data mining (Han & Kamber, 2000), the generation of representative models from data is a staple process. The models, depending on their domain of application, can be predictive or descriptive. Predictive induction has as objective the construction of a model or a set of rules to be used in classification or prediction (Chang, Lai, & Lee, 2007), while descriptive models are aimed at the discovery of individual rules which define interesting patterns in data (Yen & Lee, 2006).

Subgroup discovery (SD) is situated at the intersection of predictive and descriptive induction. In the subgroup discovery task, the rules or subgroups are discovered using heuristics which tries to find the best subgroups in terms of rule coverage and distributional unusualness (Klöesgen, 1996; Lavrač, Kavšek, Flach, & Todorovski, 2004). Subgroup discovery aims at discovering individual rules 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.

The CN2-SD (Lavrač et al., 2004) is a recent proposal in SD offering promising results. It is an adaptation of the classification rule learner CN2 algorithm based on a separate and conquer strategy (Clark & Boswell, 1989; Clark & Niblett, 1991). The main modifications are: its covering algorithm, search heuristic, probabilistic classification of instances, and evaluation measures.

The issue of scalability and the effect of increasing the size of data sets are always present in data mining (Domingo, Gavaldá, & Watanabe, 2002; Provost & Kolluri, 1999). The scaling problem, due to large size data sets, produces situations where the CN2-SD algorithm can not be executed. The evaluation necessities to apply the heuristic is expensive computationally and this cost is directly proportional to the size of the data set.

☆ This work was supported by Projects TIN2005-08386-C05-01 and TIN2005-08386-C05-03.

* Corresponding author. Tel.: +34 958 240598; fax: +34 958 243317.
E-mail addresses: jrcano@ujaen.es (J.-R. Cano), herrera@decsai.ugr.es (F. Herrera), lozano@decsai.ugr.es (M. Lozano), salvagl@decsai.ugr.es (S. García).

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.08.083

A possible way to face the scaling problem consist of scaling down the initial data sets. The scaling down can be applied by means of a pre-processing stage previous to subgroup discovery by CN2-SD. The pre-processing suggested in this paper consists in the application of data reduction techniques using instance selection algorithms (Liu & Motoda, 2002). The instance selection algorithms select representative instance subsets following a determined strategy. Those subsets composed by representative instances are used as input to extract models from them (Cano, Herrera, & Lozano, 2007; Grochowski & Jankowski, 2004; Kim, 2006; Riquelme, Aguilar, & Toro, 2003; Sebban, Nock, Chauchat, & Rakotomalala, 2000).

The aim of this paper is to propose the combination of instance selection and CN2-SD to apply the last ones into large size data sets. To address this study, we have carried out a number of experiments increasing complexity and size of data sets. We have divided the study into two parts:

- In the first one, we study the effect of instance selection in the subgroups discovered with CN2-SD in small data sets. The objective is to study if the instance selection affects the descriptive qualitative measures of the subgroups discovered.
- In the second one, we apply the instance selection in large size data sets, combined with CN2-SD and test its behaviour.

To analyze the results we provide a statistical analysis using some statistical tests (Friedman, Iman and Davenport test, Holm and Wilcoxon) which have been selected based in the considerations of Demšar in Demsar (2006). Friedman ([Friedman, 1940; Sheskin, 2000) and Iman and Davenport tests (Iman & Davenport, 1980) are nonparametric tests equivalent to the repeated-measures ANOVA (Anderson, 1984). The remaining two tests, Holm's and Wilcoxon tests (Holm, 1979; Wilcoxon, 1945), are post-hoc test that may be used only when Friedman or Iman and Davenport tests reject the null-hypothesis, under the assumption of similarity between means. Both test, Holm's and Wilcoxon tests, are used to detect significant differences between the behaviour of two algorithms.

In order to do that, the paper is set out as follows. In Section 2, we introduce the subgroup discovery task, the CN2-SD algorithm analyzed and the quality measures considered for the subgroups discovered. Section 3 is devoted to analyzing the scaling up problem which appears in the CN2-SD algorithm when large data sets are used as input. In the Section 4 we present the combination of instance selection and the subgroup discovery algorithm to face the scaling problem. Section 5 explains the methodology used in the experimentation and deals with the results and their analysis. Finally, in Section 6, we point out our conclusions.

2. Subgroup discovery

In this section we present the subgroup discovery approach. In Section 2.1 the subgroup discovery basic ideas are presented. Section 2.2 describes the subgroup discovery algorithm used in the study. Section 2.3 shows the quality measures considered for the subgroups discovered.

2.1. Description

The subgroup discovery is a task situated between the predictive and descriptive induction. It was defined by Klöesgen and Wrobel in (Klöesgen, 1996 and Wrobel, 1997) as follows: “Given a population of individuals and a property of those individuals we are interested in finding a population of subgroups that are statistically ‘most interesting’, e.g., are as large as possible and have the most unusual statistical (distributional) characteristic with respect to the property of interest”.

In subgroup discovery we are interested in the identification of relations between a dependent variable (target variable) and usually many explaining, independent variables (Lavrač, Cestnik, Gamberger, & Flach, 2004). Subgroup discovery focus its interest on partial relations instead of complete relations; (small) subgroups with interesting characteristics can be sufficient. The discovered subgroup must satisfy two conditions: They should be interpretable for the expert, and they need to be interesting according to the criteria of the user. Interestingness is typically defined by a quality function, which can take certain statistical or other user-defined quality criteria into account.

In the following, we shortly revise some subgroup discovery approaches that can be found in the specialized literature:

- Klöesgen presents the subgroup discovery task in its algorithm *Explora*, which uses divide and conquer strategy to extract the models (Klöesgen, 1996).
- Wrobel offers *Midos*, an extension of *Explora* for multi-relational data bases (Wrobel, 1997).
- Gamberger et al. propose the SD algorithm, where they introduce: a novel parametrized definition of rule quality used in a heuristic beam search algorithm, a rule subset selection algorithm incorporating example weights, the detection of statistically significant properties of selected subgroups, and a novel subgroup visualization method (Gamberger & Lavrač, 2002).
- Kavšek et al. offer the *Apriori-SD* algorithm, modifying the *Apriori-C* (which was based originally in the well-known *Apriori* algorithm for mining association rules). In this case, the classification rule discovery algorithm *Apriori-C* (Lavrač, Flach, Kavšek, & Todorovski, 2002) is adapted to subgroup discovery (Kavšek, Lavrač, & Bullas, 2002; Kavšek & Lavrač, 2006).
- Lavrač et al. present a subgroup discovery algorithm, called CN2-SD, based on the modification of CN2 classification rule learner (Clark & Boswell, 1989; Clark &

Niblett, 1991) in: its covering algorithm, search heuristic, probabilistic classification of instances, and evaluation measures [Lavrač et al., 2004].
- Železný et al. in [Železný and Lavrač, 2006] adapts relational rule learning to subgroup discovery (the algorithm is called RSD) in individual-centred domains, based on: propositionalization through first-order feature construction, feature filtering, incorporation of example weights into the weighted relative accuracy search heuristic, and implementation of the weighted covering algorithm.
- Atzmuellet et al. propose an efficient and exhaustive subgroup discovery algorithm called SD-Map, which can just be applied in two-class data sets (Atzmueller & Puppe, 2006).

Those subgroup discovery algorithms have been applied in different domains: In Gamberger, Lavrač, and Wettschereck (2002); Gamberger and Lavrač (2002), Gamberger et al. applied subgroup discovery to the problem of early detection of patient groups with risk for atherosclerotic coronary heart disease. Klöesgen et. analyze a Census data set searching interesting subgroups (Klöesgen et al., 2002). Kavšek et al. (2002) develop a case study in mining UK traffic data by means of subgroup discovery. Nakada et al. in Nakada and Kunifuji (2003) extract subgroups from personal web pages. In Gamberger and Lavrač (2002); Lavrač et al. (2004), Gambered et al. and Lavrač et. al apply expert-guided subgroup discovery for actionable knowledge generation, typically presented in the form of rules, that allows the decision maker to recognize some important relations and to perform an action, such as targeting a direct marketing campaign, or planning a population screening campaign. Lavrač studies the task of subgroup discovery in two domains in Lavrač (2005): the first one is atherosclerotic coronary heart disease and the second one is functional genomics. Berlanga et al. analyze subgroup discovery with fuzzy rules by means of a multi-objective algorithm applied to a market problem Berlanga, Del Jesus, Gonzalez, Herrera, and Mesonero (2006).

2.2. Subgroup discovery algorithm used: CN2-SD

CN2 algorithm is described at Fig. 1 (Clark & Boswell, 1989; Clark & Niblett, 1991). *CN2-SD* is based on the modification of CN2 classification rule learner in the following aspects:

- Covering task: The covering algorithm in this case is a weighted one, where the covered positive examples are not deleted from the current training set. Instead, in each run of the covering loop, the algorithm associates to each example a count which indicates how often (with how many rules) the example has been covered so far. These weights appear in the computation of WRAcc.

```
Procedure UnsortedCN2 ($all\_Examples$, $classes$)
Rules_Set ← ∅
For each class in classes
    Generate rules with OneClassCN2($all\_Examples$, $class$)
    $Rules\_Set \leftarrow Rules\_Set \bigcup rules$
Return Rules_Set

Procedure OneClassCN2 ($examples$, $class$)
rules ← ∅
Repeat
    $Best\_Condition \leftarrow Find\_Best\_Condition(examples, class)$
    If (Best_Condition is not null) then
        Add rule "if Best_Condition then class" to rules and removes all
        examples that belongs to class class covered by Best_Condition
Until Best_Condition is null
Return rule
```

Fig. 1. Pseudocode of CN2 algorithm.

- Search heuristic: The heuristic applied is the weighted relative accuracy (WRAcc). The WRAcc computation considered all probabilities computed by relative frequencies. An example of weight measures how important it is to cover this example in the next iteration. The following expression shows the way it is obtained:

$$\mathrm{WRAcc}(\mathrm{Cond} \to \mathrm{Class}) = \frac{n'(\mathrm{Cond})}{N'} \cdot \left(\frac{n'(\mathrm{Class}, \mathrm{Cond})}{n'(\mathrm{Cond})} - \frac{n'(\mathrm{Class})}{N'}\right) \tag{1}$$

where, $N^{'}$ is the sum of the weight of all examples, $n^{'}$(Cond) is the sum of the weights of all covered examples, and $n^{'}$(Class,Cond) is the sum of the weights of all correctly covered examples.
- Probabilistic classification of instances: Each CN2-SD rule returns a probability distribution, instead of class distribution in terms of the number of examples covered. Using this voting scheme the subgroups covering a small number of examples are not so heavily penalized when classifying a new example.

As the authors indicate in Kavšek and Lavrač (2006), the results offered by CN2-SD and *Apriori-SD* are very similar, but *Apriori-SD* is more sensitive when minority classes appear. CN2-SD does not present that debility.

2.3. Descriptive measures of rule interestingness

The measures of rule interestingness consider the quality of individual rules. These measures are the most appropriate for subgroup discovery, as the task of subgroup discovery is to induce individual patterns of interest. The descriptive measures considered to identify interesting rules are: *coverage*, *support*, *significance*, *unusualness*, *completeness*, and *size* (number of subgroups discovered) of the model as they are suggested by Lavrač et al. in Lavrač et al. (2004).

We have added to these one the *antecedents per rule*, that is useful for analyzing the rules. As addition we have included the *confidence* of each one of the rules to analyze their predictive behaviour. These measures evaluate each subgroup individually, but can be complemented by their variants to compute the mean of the induced set of descriptions of subgroups, allowing comparison between different subgroup discovery algorithms.

In the following, the description of each one of the measures is shown:

- Coverage:
  It is defined as the percentage of the global examples covered by one rule. Considering the rule R_i on the form Cond → Class, the expression associated to it is the following:

$$\mathrm{Cov}(R_i) = p(\mathrm{Cond}) = \frac{n(\mathrm{Cond})}{N} \tag{2}$$

  where, $n(\mathrm{Cond})$ is the number of instances where the antecedents Cond are true, and N the number of instances in the data set. The mean coverage of the rules is obtained as the expression (3) indicates:

$$\mathrm{COV} = \frac{1}{n_{\mathrm{R}}} \sum_{i=1}^{n_{\mathrm{R}}} \mathrm{Cov}(R_i) \tag{3}$$

  where n_{R} is the number of rules of the model.
- Support:
  The *support* measure computes the frequency of correctly classified covered examples of a rule R_i (Cond → Class):

$$\mathrm{Sup}(R_i) = p(\mathrm{Class}, \mathrm{Cond}) = \frac{n(\mathrm{Class}, \mathrm{Cond})}{N} \tag{4}$$

  where $n(\mathrm{Class}, \mathrm{Cond})$ is the number of instances of Class where the antecedents Cond are true. The average rule support is computed as the average Sup of all the rules which compose the model, and is defined as follows:

$$\mathrm{SUP} = \frac{1}{n_{\mathrm{R}}} \sum_{i=1}^{n_{\mathrm{R}}} \mathrm{Sup}(R_i) \tag{5}$$

- Confidence:
  To analyze the predictive capabilities of the subgroups discovered, the *confidence* of each rule is obtained. This measure represents the number of positive instances covered among all the instances covered by the rule:

$$\mathrm{Conf}(R_i) = \frac{p(\mathrm{Class}|\mathrm{Cond})}{p(\mathrm{Cond})} = \frac{n(\mathrm{Class}, \mathrm{Cond})}{n(\mathrm{Cond})} \tag{6}$$

  The mean confidence of the rules is obtained as the expression (7) indicates:

$$\mathrm{CONF} = \frac{1}{n_{\mathrm{R}}} \sum_{i=1}^{n_{\mathrm{R}}} \mathrm{Conf}(R_i) \tag{7}$$

- Significance:
  It represents how significant is a rule measured by a statistical criterion. The statistical criterion considered is the likelihood ratio of a rule, normalized with the likelihood ratio of the significance threshold (99%). Considering the rule R_i on the form Cond → Class, for each class Class_j, the expression of significance is:

$$\mathrm{Sig}(R_i) = 2 \cdot \sum_j n(\mathrm{Class}_j, \mathrm{Cond}) \cdot \log \frac{n(\mathrm{Class}_j, \mathrm{Cond})}{n(\mathrm{Class}_j)} \tag{8}$$

  where $n(\mathrm{Class}_j, \mathrm{Cond})$ is the number of instances of Class_j where the antecedents Cond are true, and $n(\mathrm{Class}_j)$ is the number of instances in the data set which belongs to Class_j. The number of different classes is j.
  We consider the average significance of the set of rules (see (9)):

$$\mathrm{SIG} = \frac{1}{n_{\mathrm{R}}} \sum_{i=1}^{n_{\mathrm{R}}} \mathrm{Sig}(R_i) \tag{9}$$

- Unusualness:
  The unusualness of the rules is defined as the weighted relative accuracy (WRAcc). This measure is a variant of rule accuracy that can be applied in the descriptive and predictive induction framework. It trades off generality of the rule ($p(\mathrm{Cond})$, i.e., rule coverage) and relative accuracy ($p(\mathrm{Class}|\mathrm{Cond}) - p(\mathrm{Class})$). It is defined as follows:

$$\begin{aligned}\mathrm{WRAcc}(R_i) &= p(\mathrm{Cond}) \cdot (p(\mathrm{Class}|\mathrm{Cond}) - p(\mathrm{Class})) \\ &= \frac{n(\mathrm{Cond})}{N} \cdot \left(\frac{n(\mathrm{Class}, \mathrm{Cond})}{n(\mathrm{Cond})} - \frac{n(\mathit{Class})}{N} \right)\end{aligned} \tag{10}$$

  where $p(\mathrm{Cond})$ is the probability of the *Condition* of the rule satisfied (rule coverage), p(Class—Cond) is the probability of the *Condition* and the *Class* satisfied and finally, $p(\mathrm{Class})$ is the probability that one instance belongs to that *Class*.
  The average rule unusualness is computed as the average WRAcc considering all the rules, is defined as:

$$\mathrm{WRACC} = \frac{1}{n_{\mathrm{R}}} \sum_{i=1}^{n_{\mathrm{R}}} \mathrm{WRAcc}(R_i) \tag{11}$$

- Completeness:
  For subgroup discovery it is interesting to compute the whole number of examples which are covered by the set of rule obtained (subgroups discovered). It is called the *completeness*, defined as the percentage of target examples (positives) covered by the rules, and computed as the true positive rate for the union of subgroups:

$$\mathrm{COMP} = \frac{1}{N} \sum_{\mathrm{Class}_j} n\left(\mathrm{Class}_j \bigvee_{\mathrm{Cond} \to \mathrm{Class}_j} \mathrm{Cond} \right) \tag{12}$$

The examples covered by several rules are counted only once. This measure is called in Lavrač et al. (2004) as overall support of a rule set.

- Size:
  The *size* is a measure that considers the number of rules which compose the model (see expression (13)). Reducing the size of the model increases the interpretability by the user.

$$\mathrm{SIZE} = n_{\mathrm{R}} \tag{13}$$

- Number of antecedents:
  To analyze the interpretability of the model we study the size of the model considering the number of rules which composed the model (*SIZE*), and the number of antecedents which compose each rule.
  Being a rule R_i in the form Cond → Class, and Cond composed by (Antecedent$_1$ ∧ Antecedent$_2$ ∧ ... ∧ Antecedent$_k$), this measure is defined as the following expression:

$$\mathrm{Ant}(R_i) = k \tag{14}$$

The average number of antecedents in the rule set is described in the expression:

$$\mathrm{ANT} = \frac{1}{n_{\mathrm{R}}} \sum_{i=1}^{n_{\mathrm{R}}} \mathrm{Ant}(R_i). \tag{15}$$

3. Analysis of the scaling problem for CN2-SD

The CN2-SD subgroup discovery algorithm is based on separate and conquer strategies. It considers a beam of rules which are generated using a determined heuristic (see Section 2.2).

The drawback it presents is that the evaluation needed to apply the heuristic is expensive computationally, and this cost is directly proportional to the size of the data set.

We study this problem using Pen Based data set. It has been obtained from the UCI Repository Newman, Hettich, Blake, and Merz, 1998, and its characteristics are shown in Table 2. It has been chosen due to the fact it is big enough to present problems for the execution of CN2-SD using the whole data set as input. We are going to split the data set and create subsets of different sizes (the percentages of instances per class are maintained), executing CN2-SD over them and studying the CN2-SD behaviour and the previous discretization process needed for its execution (the parameters fixed for CN2-SD appear in Section 5.1).

Table 1 and Figs. 2–4 show the results. Considering the Table 1, the first column is the percentage of instances kept, the number of them appears in the second column and in order of ocurrence, the time consumed by the discretization method, the CN2-SD algorithm and the combination of both (the three execution times in seconds). The figures show the evolution of the execution time of the discretization method (Fig. 2), the CN2-SD algorithm (Fig. 3) and their combined execution (Fig. 4), when the size of the input data set increases. The algorithm has been run in a Pentium 4, 3.6 Ghz, 1 Gb RAM, 320 Gb HD.

Table 1
Discretization and CN2-SD execution times in seconds increasing the size of a Pen Based subset

Data set size (%)	Number of instances	ID3-Discr. time	CN2-SD exec. time	Exec. time (ID3 Disc. + CN2-SD)
1	109	0.06	23.55	23.62
2.5	274	0.54	51.60	52.14
5	549	7.32	210.88	218.20
7.5	824	32.83	1312.49	1345.30
10	1105	78.14	1664.40	1792.54
20	1648	121.01	4632.19	4753.20
30	3297	519.96	29525.63	30045.59
40	4396	911.43	72605.73	73517.16
50	5496	1554.63	121435.09	122989.72

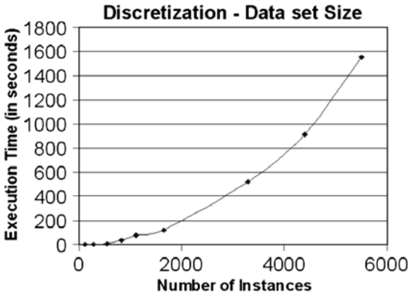

Fig. 2. Scaling evolution of the ID3 discretization method using subsets of Pen Based as input.

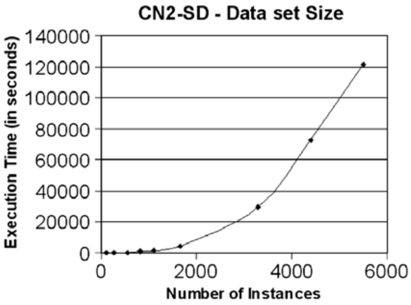

Fig. 3. Scaling evolution of the CN2-SD algorithtm using subsets of Pen Based as input.

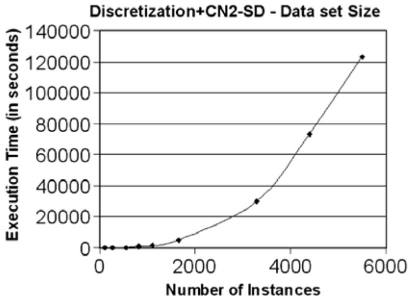

Fig. 4. Scaling evolution of the discretization and CN2-SD algorithtm using subsets of Pen Based as input.

The results which appear in Table 1 and Figs. 2–4 show that the execution time needed by CN2-SD when the size of data set increases (due to its own execution time and its

Table 2
Discretization + CN2-SD executions with different size data sets

Data sets	# Instances	# Attributes	# Classes	Execution time (s)
Iris	150	4	3	1.30
Lymphography	148	18	4	16.53
Wine	178	13	3	31.42
Led24Digit	200	24	10	47.55
Pima	768	8	2	49.56
Glass	294	9	7	60.38
Contraceptive	1473	10	3	1106.72
Pen Based	10,992	16	10	Not runnable
Adult	45,221	14	2	Not runnable

discretization method associated) makes it difficult to be used for large size data sets.

As second study we execute CN2-SD using different size data sets. These data sets are extracted from the UCI Repository Newman et al., 1998. Table 2 presents the characteristics of the data sets and the mean execution time for a ten fold cross validation for CN2-SD on them. It contains by columns the name of the problem, its number of instances, number of features and number of classes and finally, the average execution time of CN2-SD and its discretization method (in seconds). Like in the previous study, the computer used is a Pentium 4, 3.6 Ghz, 1 Gb RAM, 320 Gb HD.

Table 2 shows that when the size of the input data set increases, the execution time and resources needs for the execution increase too, so for large data sets the CN2-SD cannot be executed. Pen Based and Adult data sets introduce in CN2-SD problems due to memory consumption and elevated execution time.

4. Instance selection pre-processing for CN2-SD subgroup discovery in large size data sets

When the input data set size affects the execution of the algorithms, we can face this situation following two different strategies:

- Scaling up the algorithm. Proposing faster and lower consumption algorithms that can face large size data sets.
- Scaling down the data set. In this case, the attention is directed toward the data set. The idea consists of modifying the data set by mean of reductions to make it adequate for the original algorithm.

In this paper we pay attention to the second strategy. We are interested in the application of a pre-processing stage to reduce the initial data set previously to the model extraction. The reduction of the initial data set can be developed following different paths: instance selection Liu and Motoda, 2002, 2003; Wilson and Martinez, 2000, feature selection (Polat and Gunes, 2007; Shang et al., 2006; Huang and Wang, 2006, or data generation (Sánchez, 2004). In this study we are interested in the reduction using instance selection algorithms.

In instance selection we want to isolate the smallest set of instances which enable us to predict the class of a query instance with the same quality as the initial data set (Liu & Motoda, 2001; Liu & Motoda, 2002). By reducing the 'useful' data set size we can reduce the space complexity and decrease computational cost of the data mining algorithms that will be applied later, improving their generalization capabilities due to the elimination of noise.

As instance selection algorithms we have selected for this study those which show the best behaviour in Cano, Herrera, and Lozano (2003), with low resources consumption and high reduction rates:

- CNN (Hart, 1968) – It tries to find a consistent subset, which correctly classifies all of the remaining points in the sample set. The CNN algorithm finds a subset S of the training set TR such that every member of TR is closer to a member of S of the same class than to a member of S of a different class. The subset S can be used to classify all the instances in TR correctly. A description of the algorithm is given at Fig. 5.
- IB2 (Kibbler & Aha, 1987) – It is similar to CNN but using a different selection strategy. The difference with CNN is that IB2 does not seed S with one instance of each class, and does not repeat the process after the first pass through the training set. This means that IB2 will not necessarily classify all instances in TR correctly. The algorithm retains border points in S while eliminating internal points that are surrounded by members of the same class. The pseudo code of the algorithm is offered in Fig. 6.
- IB3 (Kibbler & Aha, 1987) – Instance x from the training set TR is added to the new set S if the nearest *acceptable* instance in S (if there are not *acceptable* instances a random one is used) has different class than x. The *acceptable* concept is defined as the confidence interval:

$$\frac{p + \frac{z^2}{2n} \pm z\sqrt{\frac{p(p-1)}{n} + \frac{z^2}{2n^2}}}{1 + \frac{z^2}{n}} \tag{16}$$

```
$TR = Examples\_Set$, $S = \emptyset$, $fail = true$
$S = S \bigcup \{x_{C_1}, x_{C_2}, \ldots, x_{C_M}\}$, where $x_{C_i}$ is any example that belongs to class $i$
While $fail = true$
    $fail = false$
    For each example $x$ in $TR$
        If $x$ is misclassified by using $S$
            $S = S \bigcup \{x\}$
            $fail = true$
Return $S$
```

Fig. 5. Pseudocode of CNN algorithm.

```
1. $TR = Examples\_Set$, $S = \emptyset$, $fail = true$
2. For each example $x$ in $TR$
    3. If $x$ is misclassified by using $S$
        4. $S = S \bigcup \{x\}$
5. Return $S$
```

Fig. 6. Pseudocode of IB2 algorithm.

z is the confidence factor (0.9 is used to accept, 0.7 to reject). *p* is the classification accuracy of a *x* instance (while *x* is added to *S*). *n* is the number of classification-trials for given instance (while added to *S*). The algorithm proceeds as shown in Fig. 7.

- Drop3 (Wilson & Martinez, 1997) – It uses a noise filtering pass before sorting the instances in TR. This is done using the rule: Any instance not classified by its *k*-nearest neighbours is removed. After removing noisy instances from *S* in this manner, the instances are sorted by distance to their nearest enemy remaining in *S*, and thus points far from the real decision boundary are removed first. This allows points internal to clusters to be removed early in the process, even if there were noisy points nearby. After the noise removal, the steps of the algorithms are similar to DROP2 (Wilson & Martinez, 1997), and are described in Fig. 8.
- ICF (Brightom & Mellish, 2002) – ICF defines Reachability(*x*) and *Coverage*(*x*) sets. In the first stage, ICF employs ENN algorithm [Wilson, 1972] to remove noisy sample from *T*. Then, in second stage, it removes each instance *x* for which the Reachability(*x*) is bigger than the Coverage(*x*). It recalculates reachability and coverage properties and restarts the second stage all many times as possible. The algorithm proceeds as shown in Fig. 9.
- Evolutionary instance selection based on CHC algorithm (EIS-CHC) (Cano et al., 2003; Eshelman, 1991) – Evolutionary algorithms (Back, Fogel, & Michalewicz, 1997) are general-purpose search algorithms that use principles inspired by natural genetic populations to evolve solutions to problems, and they have been used to solve the instance selection problem, with promising

```
1. For each instance $x$ in $TR$
2. Let $a$ be the nearest acceptable instance in $S$ to $x$ (if there are no
acceptable instances in $S$, let $a$ be a random instance in $S$)
3. If class(a)$\neq$class(x) then add $x$ to $S$.
4. For each instance $s$ in $S$
    5. If $s$ is at least as close to $x$ as $a$ is
        6. Then update the classification record of $s$ and remove $s$ from $S$
            if its classification record is significantly poor
7. Remove all non-acceptable instance from $S$
8. Return $S$
```

Fig. 7. Pseudocode of IB3 algorithm.

```
1. Let $S = TR$
2. For each instance $s$ in $S$
    3. Find $s$.N1..k+1, the k+1 nearest neighbors of $s$ in $S$
    4. Add $s$ to each of its neighbors lists of associates
5. For each instance $s$ in $S$
    6. Let $with$ = $\sharp$ of associates of $s$ classified correctly with $s$ as a neighbor
    7. Let $without$ = $\sharp$ of associates of $s$ classified correctly without $s$
    8. If (without - with) = 0
        9. Remove $s$ from $S$ if at least as many of its associates in $TR$ would
        be classified correctly without $s$.
        10. For each associate $a$ of $s$ many
            11. Remove $s$ from $as$ list of nearest neighbors
            12. Find a new nearest neighbor for $a$
            13. Add $a$ to its new neighbors list of associates
        14. For each neighbor $k$ of $s$
            15. Remove $s$ from $ks$ lists of associates
16. Return $S$
```

Fig. 8. Pseudocode of DROP3 algorithm.

```
ICF(T)
1. Perform Wilson Editing
2. For all $x \in T$ do
    3. If $x$ classified incorrectly by k nearest neighbours then
        4. Flag $x$ for removal
5. For all $x \in T$ do
    6. If $x$ flagged for removal then $T$=$T$-$x$
7. Iterate until no cases flagged for removal:
8. Repeat
    9. For all $x \in T$ do
        10. Compute reachable($x$)
        11. Compute coverage($x$)
    12. Progress=false
    13. For all $x \in T$ do
        14. If | reachable($x$) | > | coverage($x$) | then
            15. Flag $x$ for removal
            16. Progress=true
    17. For all $x \in T$ do
        18. If $x$ flagged for removal then $T$=$T$-$x$
19. Until not Progress
20. Return $T$
```

Fig. 9. Pseudocode of ICF algorithm.

results (Kuncheva, 1995; Kim, 2006). The election of CHC as instance selection algorithm is based in its behaviour showed on (Cano et al., 2003; Cano, Herrera, & Lozano, 2005).
During each generation the EIS-CHC develops the following steps:

(1) It uses a parent population of size pop to generate an intermediate population of pop individuals, which are randomly paired and used to generate pop potential offspring.
(2) Then, a survival competition is held where the best pop chromosomes from the parent and offspring populations are selected to form the next generation.

Other important characteristics of this algorithm are:

- CHC also implements a form of heterogeneous recombination using HUX, a special recombination operator. HUX exchanges half of the bits that differ between parents, where the bit position to be exchanged is randomly determined. CHC also employs a method of incest prevention. Before applying HUX to two parents, the Hamming distance between them is measured. Only those parents who differ from each other by some number of bits (mating threshold) are mated. The initial threshold is set at $L/4$, where L is the length of the chromosomes. If no offspring are inserted into the new population then the threshold is reduced by 1.
- No mutation is applied during the recombination phase. Instead, when the population converges or the search stops making progress (i.e., the difference threshold has dropped to zero and no new offspring are being generated which are better than any members of the parent population) the population is reinitialized to introduce new diversity to the search. The chromosome representing the best solution found over the course of the search is used as a template to re-seed the population. Re-seeding of the population is accomplished by randomly changing 35% of the bits in the template chromosome to form each of the other n-1 new chromosomes in the population. The search is then resumed.

The instance selection algorithms are also affected by the size of the input data set (Cano et al., 2005). The effect it produces on them are:

- Efficiency, due to the fact that instance selection algorithms present execution orders greater than $O(n^2)$.
- Resources, in the sense that most of the instance selection algorithms need the complete data set stored in memory to carry out their execution.
- Representation, in the case of evolutionary instance selection, the large size of chromosome used to represent the solution produces convergence difficulties for the algorithm.

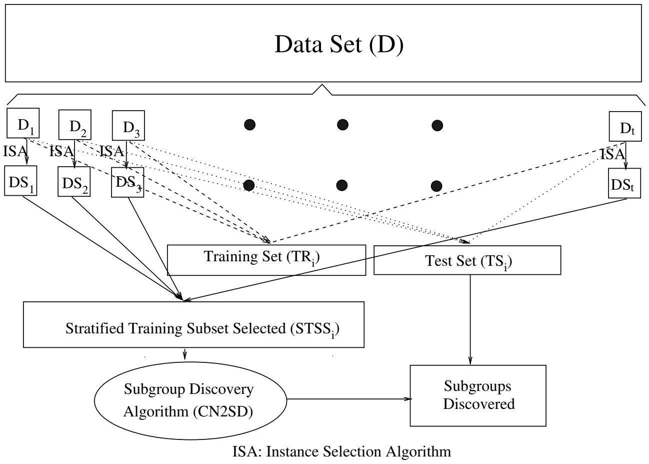
Data Set (D)
D1
D2
D3
Dt
ISA
ISA
ISA
ISA
DS1
DS2
DS3
DSt
Training Set (TRi)
Test Set (TSi)
Stratified Training Subset Selected (STSSi)
Subgroup Discovery Algorithm (CN2SD)
Subgroups Discovered
ISA: Instance Selection Algorithm

Fig. 10. Instance selection strategy in stratified ten fold cross validation for subgroup discovery.

To avoid the drawbacks associated to large size data sets we apply the instance selection combined with stratified strategy as it was suggested in Cano et al. (2005) with promising results.

Following the stratified strategy, initial data set D is divided into t disjoint sets D_j, strata of equal size, D_1, $D_2, \ldots,$ and D_t.

The test set (TS) will be the complementary one in D to the training set (TR). The subsets TR and TS will be obtained as (17) and (18) show:

$$\mathrm{TR} = \bigcup_{j \in J} D_j, J \subset \{1, 2, \ldots, t\} \tag{17}$$

$$\mathrm{TS} = D \setminus \mathrm{TR} \tag{18}$$

Instance selection algorithms are applied in each D_j obtaining a subset selected DS_j. The instance selected set (TSS) in stratified strategy is obtained using the DS_j (see Eq. (19)) and it is called Stratified Training Subset Selected (STSS).

$$\mathrm{STSS} = \bigcup_{j \in J} \mathrm{DS}_j, J \subset \{1, 2, \ldots, t\} \tag{19}$$

The complete process is presented in Fig. 10.

5. Experimental study

As we have mentioned, we divide our study into two parts:

- The first one studies if the instance selection affects the descriptive quality measures of the subgroups discovered.
- In the second part, we apply the CN2-SD subgroup discovery algorithm in large size data sets, combined with instance selection algorithms.

In order to develop that, this section is organized as follows. In the Subsection 5.1 we present the algorithms, parameters and data sets considered. The Subsection 5.2 describes the non-parametric statistical procedures used for analyzing the results obtained. The Subsection 5.3 is dedicated to the first study. In the Subsection 5.4 the second study is offered. An illustrative example of extraction of rules is shown in Section 5.5.

5.1. Algorithms, parameters and data sets

The experimental methodology is defined in three aspects: Data sets, algorithms and parameters. They are as follows:

- Data sets: In the first study we consider the small size data sets (Lymphography, Iris, Wine, Led24Digit, Glass, Pima and Contraceptive) and in the second one the large size ones (Pen-Based and Adult). Pen-Based is divided in 10 strata and Adult in 100 to execute the instance selection algorithms. The characteristics of the data sets appear in the Table 2.
- Algorithms: CN2-SD, CNN + CN2-SD, IB2 + CN2-SD, IB3 + CN2-SD, DROP3 + CN2 − SD, ICF + CN2-SD and EIS − CHC + CN2 − SD, described in Section 2.2 for the first one and Section 4 for the rest.
- Parameters: The parameters are chosen considering the authors suggestions in the literature. For each one of the algorithms are:
  - CN2-SD: beam-size = 5, significance-threshold = 99% and multiplicative weights with $\gamma = 0.9$. The selection of the cut points for the numeric antecedents of the rules have been done using a discretization method, concretely the ID3 discretization method (Liu, Hussain, Tan, & Dash, 2002) based on entropy (Janssens, Brijs, Vanhoof, & Wets, 2006).
  - CNN: It has not parameters to be fixed.
  - IB2: It has not parameters to be fixed.
  - IB3: Acceptance level = 0.9 and drop level = 0.7.
  - DROP3: It has not parameters to be fixed.
  - ICF: It has not parameters to be fixed.
  - EIS-CHC: Evaluations = 10,000, population = 50 and $\alpha = 0.5$.

The deterministic algorithms have been executed one time for each partition in the ten fold cross validation and three times the non-deterministic ones. The results obtained for the size and number of antecedents indexes consider the subgroups (rules) extracted from the training set. The rest of the measures are evaluated over the test data set. The table of results for each one of the data sets appears in the Appendix A.

5.2. On the use of non-parametric statistical procedures for analyzing the results

We are interested in the study of the effect of instance selection algorithms in the discovered subgroups. For this reason we apply instance selection in small size data sets and analyse the quality indexes of the subgroups discovered by CN2-SD. In this case, due to the size of the data sets, we do not need the stratification, using the whole training set.

The results for each one of the small data sets, considering the mean of the measures, are showed in the appendix. To compare the results provided by CN2-SD over the different training set selection algorithm outputs we develop a statistical analysis using the executions per algorithm for each measure. Statistical analysis have carried out in order to find significant differences among the results obtained by the studied methods. When a parametric test is used, results must assume normal distribution and homogeneity of variance. If these assumptions are satisfied, a parametric statistical analysis of results will be right and safe.

In our situation, we consider the use of non-parametric tests, according to the recommendations made in Demsar (2006).

As such, these non-parametric tests can be applied to classification accuracies, error ratios or any other measure for evaluation of techniques, including even model sizes and computation times. Empirical results suggest that they are also stronger than the parametric test. Demšar recommends a set of simple, safe and robust non-parametric tests for statistical comparisons of classifiers. We will present the main tests with different purposes.

- The first one is the Friedman test Friedman (1940); Sheskin (2000), which is a non-parametric test equivalent of the repeated-measures ANOVA. Under the null-hypothesis, it states that all the algorithms are equivalent, so a rejection of this hypothesis implies the existence of differences among the performance of all the algorithms studied. After this, a post-hoc test could be used in order to find whether the control or proposed algorithm presents statistical differences with regards to the remaining methods in the comparison. The simplest of them is the Bonferroni–Dunn test, but we can use more powerful test that controls the family-wise error and rejects more hypothesis than Bonferroni–Dunn test; for example, Holm's test.

  Friedman test way of working is described as follows: It ranks the algorithms for each data set separately, the best performing algorithm getting the rank of 1, the second best rank 2, and so on. In case of ties average ranks are assigned.

  Let r_i^j be the rank of the jth of k algorithms on the ith of N_{ds} data sets. The Friedman test compares the average ranks of algorithms, $R_j = \frac{1}{N_{\text{ds}}}\sum_i r_i^j$. Under the null-hypothesis, which states that all the algorithms are equivalent and so their ranks R_j should be equal, the Friedman statistic:

$$\chi_F^2 = \frac{12N_{\text{ds}}}{k(k+1)}\left[\sum jR_j^2 - \frac{k(k+1)^2}{4}\right] \qquad (20)$$

is distributed according to χ_F^2 with $k-1$ degrees of freedom, when N_{ds} and k are big enough (as a rule of a thumb, $N_{ds} > 10$ and $k > 5$).

- The second one of them is the Iman and Davenport test (Iman & Davenport, 1980), which is a non-parametric test, derived from the Friedman test, less conservative than the Friedman statistic:

$$F_F = \frac{(N_{ds}-1)\chi_F^2}{N_{ds}(K-1)-\chi_F^2} \tag{21}$$

which is distributed according to the F-distribution with $k-1$ and $(k-1)(N_{ds}-1)$ degrees of freedom. Statistical tables for critical values can be found at (Sheskin, 2000; Zar, 1999).

- As post-hoc test of Friedman statistic, we will use the Holm test (Holm, 1979), which is a multiple comparison procedure that works with a control algorithm (normally, the best of them is chosen) and compares it with the remain of methods. The test statistics for comparing the ith and jth method using this procedure is:

$$z = (R_i - R_j) \Big/ \sqrt{\frac{k(k+1)}{6N_{ds}}} \tag{22}$$

The z value is used to find the corresponding probability from the table of normal distribution, which is then compared with an appropriate α. In Bonferroni–Dunn comparison, this α value is always $\alpha/(k-1)$, but Holm's test adjust the value for α in order to compensate for multiple comparison and control the family-wise error. Holm's test is a step-up procedure that sequentially tests the hypothesis ordered by their significance. We will denote the ordered p values by $p_1, p_2, \ldots,$ so that $p_1 \leqslant p_2 \leqslant \cdots \leqslant p_{k-1}$. Holm's test compares each p_i with $\alpha/(k-i)$, starting from the most significant p value. If p_1 is below $\alpha/(k-1)$, the corresponding hypothesis is rejected and we allow to compare p_2 with $\alpha/(k-2)$. If the second hypothesis is rejected, the test proceeds with the third, and so on. As soon as a certain null hypothesis cannot be rejected, all the remaining hypothesis are retained as well.

- Finally, we will describe the Wilcoxon Signed-Ranks Test (Wilcoxon, 1945; Sheskin, 2000): This is the analogous of the paired t-test in non-parametrical statistical procedures; therefore, it is a pair wise test that aims to detect significant differences between the behaviour of two algorithms. In our study, we always consider a level of significance of $p < 0.05$.

Let d_i be the difference between the performance scores of the two classifiers on ith out of N_{ds} data sets. The differences are ranked according to their absolute values; average ranks are assigned in case of ties. Let R^+ be the sum of ranks for the data sets on which the second algorithm outperformed the first, and R^- the sum of ranks for the opposite. Ranks of $d_i = 0$ are split evenly among the sums; if there is an odd number of them, one is ignored:

$$R^+ = \sum_{d_i>0} \mathrm{rank}(d_i) + \frac{1}{2}\sum_{d_i=0} \mathrm{rank}(d_i) \tag{23}$$

$$R^- = \sum_{d_i<0} \mathrm{rank}(d_i) + \frac{1}{2}\sum_{d_i=0} \mathrm{rank}(d_i) \tag{24}$$

Let T be the smallest of the sums, $T = \min(R^+, R^-)$. In this work, we have used seven data sets to carry out the study. The critical value associated with $N_{ds} = 7$ can be found in the T Wilcoxon distribution (see table B.12 in Zar (1999)) considering 7 degrees of freedom. The request critical value for this work is equal to 2.

The Wilcoxon signed ranks test is more sensitive than the t-test. It assumes commensurability of differences, but only qualitatively: greater differences still count more, which is probably desired, but the absolute magnitudes are ignored. From the statistical point of view, the test is safer since it does not assume normal distributions. Also, the outliers (exceptionally good/bad performances on a few data sets) have less effect on the Wilcoxon than on the t-test. The Wilcoxon test assumes continuous differences d_i, therefore they should not be rounded to, say, one or two decimals since this would decrease the power of the test due to a high number of ties.

When the assumptions of the paired t-test are met, the Wilcoxon signed-ranks test is less powerful than the paired t-test. On the other hand, when the assumptions are violated, the Wilcoxon test can be even more powerful than the t-test.

5.3. Instance selection combined with CN2-SD in small size data sets

Considering the description of the statistical procedures in the previous section, we present in Table 3 the ranking of the algorithms evaluated using the seven small size data sets for each one of the quality indexes.

With these ranks, the Friedman and Iman–Davenport tests are applied for every index considering the null-hypothesis, whose acceptation means the non-existence of differences among the indexes obtained for the algorithms. The rejection means that differences appear. Table 4 shows the response of both tests.

In both tests the response is the same. They do not find any difference in COB, COMP, SIG, SUP and WRACC. This means that instance selection maintains the quality of those indexes. The differences appear in ANT, CONF and SIZE.

To analyze these differences we have to apply a post-hoc test. As post-hoc tests of Friedman statistic we have introduced the Holm's test to find whether control algorithm presents statistical differences with the rest of the algorithms. The control algorithm considered for each index is the one with the highest rank for that index in the Table 3. Indeed, Tables from 5–7 contain the computation of data performed through the Holm procedure for detecting

Table 3
Average rank for the algorithms considering all measures

Measure	CN2-SD	CNN + CN2-SD	DROP3 + CN2-SD	IB2 + CN2-SD	IB3 + CN2-SD	ICF + CN2-SD	CHC + CN2-SD
ANT	4.714	4.429	4.286	3.857	5.571	4.143	1.000
COB	2.714	4.000	5.286	4.000	3.429	4.571	4.000
COMP	3.214	4.000	3.786	4.214	3.643	3.571	5.571
CONF	4.571	5.286	2.571	5.000	4.714	2.286	3.571
SIG	4.143	3.857	3.571	5.143	4.714	3.714	2.857
SUP	2.429	5.143	4.714	5.143	3.714	3.571	3.286
WRACC	5.143	3.714	4.000	3.000	4.143	4.429	3.571
SIZE	6.429	4.571	3.786	4.500	4.714	3.000	1.000

Table 4
Multiple comparison tests results ($p = 0.05$)

Measure	Friedman statistic	χ^2 critical value	Hyp.	Iman–Davenport statistic	F critical value	Hyp.
ANT	18.429	12.592	Reject	4.691	2.360	Reject
COB	5.939	12.592	Accept	0.998	2.360	Accept
COMP	5.235	12.592	Accept	0.854	2.360	Accept
CONF	12.980	12.592	Reject	2.684	2.360	Reject
SIG	5.143	12.592	Accept	0.837	2.360	Accept
SUP	9.551	12.592	Accept	1.766	2.360	Accept
WRACC	4.163	12.592	Accept	0.660	2.360	Accept
SIZE	25.546	12.592	Reject	9.315	2.360	Reject

Table 5
Holm table for ANT

i	Algorithm	$z = (R_0 - R_i)/SE$	p	α/i	Hyp.
6	Ib3 + CN2-SD	3.958973274443149	7.527265841736313E−5	0.0083	Reject
5	CN2 + SD	3.216665785485058	0.0012968957989590842	0.0100	Reject
4	CNN + CN2-SD	2.9692299558323616	0.00298547089126885	0.01250	Reject
3	Drop3 + CN2-SD	2.845512041006012	0.004434008303100685	0.0167	Reject
2	ICF + CN2-SD	2.721794126179664	0.0064928577450838855	0.0250	Reject
1	Ib2 + CN2-SD	2.4743582965269675	0.01334757592684311 8	0.0500	Reject

Control algorithm: EIS-CHC + CN2-SD.

Table 6
Holm table for CONF

i	Algorithm	$z = (R_0 - R_i)$/SE	p	α/i	Hyp.
6	CNN + CN2-SD	2.598076211353316	0.009374768459434853	0.0083	Accept
5	Ib2 + CN2-SD	2.3506403817006194	0.018741136789596654	0.0100	Accept
4	Ib3 + CN2-SD	2.103204552047923	0.03544789255246077	0.0125	Accept
3	CN2-SD	1.979486637221574	0.04776124267510374	0.0167	Accept
2	EIS-CHC + CN2-SD	1.1134612334371354	0.2655103889538738	0.0250	Accept
1	Drop3 + CN2-SD	0.24743582965269667	0.8045709480174359	0.0500	Accept

Control algorithm: ICF + CN2-SD.

Table 7
Holm table for SIZE

i	Algorithm	$z = (R_0 - R_i)$/SE	p	α/i	Hyp.
6	CN2-SD	4.701280763401239	2.5853474452390883E−6	0.0083	Reject
5	Ib3 + CN2-SD	3.2166657854850573	0.0012968957989590879	0.0100	Reject
4	CNN + CN2-SD	3.09294787065871	0.0019817894378099583	0.0125	Reject
3	Ib2 + CN2-SD	3.0310889132455356	0.002436734808989053	0.0167	Reject
2	Drop3 + CN2-SD	2.4124993391137934	0.015843566166002118	0.0250	Reject
1	ICF + CN2-SD	1.7320508075688774	0.08326451666355035	0.0500	Accept

Control algorithm: EIS-CHC + CN2-SD.

the differences between the control algorithm and the remaining ones.

Studying the Holm's test tables, we can point out the following:

- Considering ANT index (Table 5):
  The comparison of EIS – CHC + CN2 – SD with the rest of the methods rejects the hypothesis that the algorithms are equivalent. The use of EIS-CHC algorithm previous the CN2-SD execution improves the ANT index.
- Considering SIZE index (Table 7):
  In the case of SIZE index, the hypothesis is rejected. The test finds differences between the control algorithm (the EIS – CHC + CN2 – SD which offers the best behaviour in this index in the ranking table) and most of the algorithms. When the hypothesis is rejected means that the control algorithm improves the other algorithm in the comparison. Just in the comparison with ICF + CN2 – SD the hypothesis is accepted, so the conclusion is that both present the same behaviour.
- Considering the remaining indexes, no differences have been found among them. This implies that these indexes are not affected by the instance selection algorithms. Note that in the case of the CONF index, multiple comparisons tests used (Friedman and Iman–Davenport) consider that there exist differences among the results, but they are not remarkable due to Holm's procedure accepts all hypotheses of comparisons among all the algorithms (Table 6), indicating their similarity in CONF index.

In resume we can indicate that the instance selection does not affect negatively the quality indexes of the subgroup discovered. They maintain most of the quality indexes and in the ones with different behaviour, like in SIZE and ANT, the EIS – CHC + CN2 – SD improves those indexes, generating smaller and more interpretable set of subgroups.

To complete the analysis of the instance selection algorithms combined with CN2-SD we include the Wilcoxon test between the instance selection algorithm with best indexes, EIS – CHC + CN2 – SD and the CN2-SD without previous reduction of the input data set.

In Table 8 we present the results of this test where we compare every index between both algorithms.

As we can see, the situation is similar than the test in the previous sections offered to us. EIS-CHC presents the same behaviour in most of the quality indexes, improving the ones related to the size and the number of subgroups (SIZE and ANT), reducing and making them more interpretable by the user.

At this point, we have studied the effect of the instance selection in the quality indexes of the subgroups discovered. Now, we are interested in the effect of the instance selection in the CN2-SD execution, concretely in its run-time.

Table 8
Wilcoxon test for a pairwise comparison between CN2-SD and EIS-CHC + CN2-SD ($p = 0.05$)

Measure	$T = \min(R^+, R^-)$	C. value	Hyp.	Best
ANT	0.0	2.0	Reject	CHC + CN2-SD
COB	13.0	2.0	Accept	–
COMP	13.0	2.0	Accept	–
CONF	8.0	2.0	Accept	–
SIG	6.0	2.0	Accept	–
SUP	13.0	2.0	Accept	–
WRACC	10.0	2.0	Accept	–
SIZE	0.0	2.0	Reject	CHC + CN2-SD

Table 9
Execution time in seconds in small size data sets for instance selection algorithms and CN2-SD after instance selection

Algorithm	Iris		Led24Digit		Contraceptive	
	Inst.Sel	CN2-SD	Inst.Sel	CN2-SD	Inst.Sel	CN2-SD
CN2-SD tfcv		1.30		41.50		1106.72
CNN + CN2-SD tfcv	0.3	0.26	0.1	28.08	0.1	140.44
IB2 + CN2-SD tfcv	0.1	0.30	0.1	28.33	0.1	168.53
IB3 + CN2-SD tfcv	0.1	0.39	0.1	27.55	0.3	157.94
DROP3 + CN2-SD tfcv	0.3	0.38	0.5	9.90	0.4	16.82
ICF + CN2-SD tfcv	0.3	0.42	0.5	10.83	0.5	27.58
EIS-CHC + CN2-SD tfcv	0.8	0.22	11.9	6.89	88.2	0.61

Table 9 shows the effect of data reduction in the CN2-SD execution. It offers the run-time for the instance selection algorithms and CN2-SD after instance selection in three different size data sets.

The reduction in run-time is remarkable when the instance selection is applied. Using any of the instance selection algorithms the reduction in execution time for CN2-SD is at least of 85%.

In conclusion to the study of instance selection combined with CN2-SD in small size data sets we make the following analysis:

- The reduction applied in the initial data set by means of instance selection algorithms maintains the quality measures of the subgroup discovered. The statistical analysis does not find differences among the algorithm in most of the measures, and when the differences appear (SIZE and ANT indexes) is in favor of instance selection algorithms.
- The models discovered are smaller (SIZE) when the preprocessing stage is developed, which makes them more interpretable. Taking note to the Contraceptive data set in Table 11 we can see the reduction in size from 56.0 with 3.4 antecedents per rule without reduction to 4.2 with 1.73 antecedents using EIS-CHC.

- Paying attention to the run-time, we can point that the instance selection reduces the run-time of CN2-SD. For example, in Contraceptive data set, from 1106.72 to 0.61 s by means of EIS-CHC algorithm, using an acceptable time for the reduction run (88.2 s).

We can reach as conclusion that the instance selection affects positively to subgroup discovery, keeping WRACC and SIG quality indexes and reducing the size of the models and the execution time of the subgroup discovery process. This conclusion lead us to propose the use of instance selection for high size data sets, allowing to subgroup discovery algorithms to extract subgroups/rules from large data sets.

5.4. Instance selection for CN2-SD subgroup discovery in large size data sets

In this section we want to apply the CN2-SD in large size data sets by means of instance selection, due to the subgroup discovery algorithm cannot be applied directly. In this section we use the stratified execution of each one of the instance selection algorithms in large size data sets, and we develop subgroup discovery over the selected subsets.

To complete the analysis we include a new Table 10 where we can study the reduction effect of instance selection algorithms on the original large size data sets. Table 10 contains the average number of instances selected by the instance selection algorithms when they are applied following the stratified strategy for Pen-Based and Adult data sets.

With these results in mind, and considering that CN2-SD presents execution problems with large size data sets, we select the instance selection algorithms with smallest subsets selected (subsets with less than 1000 instances, like EIS-CHC and IB2) to analyze their behaviour extracting subgroups in Pen-Based and Adult data sets.

Tables 11 and 12, for Pen-based and Adult data sets respectively, contain the results obtained with the combination of instance selection and CN2-SD, providing results on all of the indexes considered and the time consumed by the CN2-SD execution in seconds.

Analyzing Tables 11 and 12, we can point out the following conclusions:

- The execution of CN2-SD in Adult data set using the subset selected by the algorithm IB2 Strat is composed by 594.2 instances and it takes 2104.571 s and 5245.769 s for running the subset selected for EIS-CHC. This situation indicates the difficulties which appear to evaluate the CN2-SD over the whole data set.
- In the medium size data set (Pen Based) the results are similar between IB2 and EIS-CHC, but when the size increases, the last one presents the best behaviour. It offers the subgroups with highest SIG, WRACC, CONF, SUP and COMP, and the number of subgroup (SIZE) is smaller than the obtained by IB2.

As conclusion, we point out that the combination of the highest reduction instance selection algorithms let us to run the CN2-SD subgroup discovery algorithm in large size data sets.

5.5. Analysis of the rules extracted in large size data sets: Adult data set

In this section we present some of the rules extracted from the adult data set, the largest one used in this study. The instance selection method used is the EIS-CHC, due to its good behaviour in the previous section.

Table 10
Number of instances selected in mean by the instance selection algorithms executed using the stratified strategy

Data set	Original size	CNN	IB3	EIS-CHC	IB2	DROP3	ICF
Pen-Based	10992	1003.2	1043.5	332.3	603.2	32.4	2498.3
Adult	45221	17407.4	11734.1	896.8	594.2	1904.6	7352.1

Table 11
Stratified instance selection in Pen-Based data set

Algorithm	SIG	WRACC	SIZE	ANT	COB	CONF	SUP	COMP	TIME
IB2 Strat. + CN2-SD	162.614	0.057	42.5	3.575	0.225	0.404	0.079	0.999	301.6
CHC Strat. + CN2-SD	145.272	0.053	25.8	3.562	0.188	0.467	0.072	0.997	356.4

Table 12
Stratified instance selection in adult data set

Algorithm	SIG	WRACC	SIZE	ANT	COB	CONF	SUP	COMP	TIME
IB2 Strat. + CN2-SD	102.548	0.045	22.3	3.271	0.290	0.429	0.179	0.998	2104.571
CHC Strat. + CN2-SD	159.120	0.066	16.1	3.385	0.415	0.759	0.330	0.992	5245.769

In adult data set the instances which compose it are obtained from the US Census and the prediction task desired consist of on classifying when one person has a salary higher or lower than 50,000. It is available in the UCI Repository (Newman et al., 1998), and its characteristics appear in Table 2.

The information considered of each one of those people and the domains of their values are:

- Age: Continuous value.
- Workclass: Private, self-emp-not-inc, self-emp-inc, federal-gov, local-gov, state-gov, without-pay, never-worked.
- Final-weight: Continuous.
- Education: Preschool, 1st–4th, 5th–6th, 7th–8th, 9th, 10th, 11th, 12th, HS-grad, some-college, assoc-voc, assoc-acdm, bachelors, masters, prof-school, doctorate.
- Education-num: Continuous. This attribute is a numerical representation of the previous education attribute. We keep it to conserve the original adult data set. The association between this attribute and the previous one is the order of the list in which the value appears (from 1 in preschool case to 16 in doctorate one).
- Marital-status: Married-civ-spouse, divorced, never-married, separated, widowed, married-spouse-absent, married-AF-spouse.
- Occupation: Tech-support, craft-repair, other-service, sales, exec-managerial, prof-specialty, handlers-cleaners, machine-op-inspct, adm-clerical, farming-fishing, transport-moving, priv-house-serv, protective-serv, armed-forces.
- Relationship: Wife, own-child, husband, not-in-family, other-relative, unmarried.
- Race: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
- Sex: Female, male.
- Capital-gain: Continuous.
- Capital-loss: Continuous.
- Hours-per-week: Continuous.
- Native-country: United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US (Guam-USVI-etc.), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinidad and Tobago, Peru, Hong, Holland-Netherlands.

The possible values for prediction are: ⩽50K, >50K.

We present some of the rules extracted for each one of the two classes, showing their quality indexes.

Some rules for the class *person who earn* ⩽50K:

(1) IF education-num ⩽ 11.5 AND capital-loss ⩽1784.0 AND capital-gain ⩽5100.0 THEN Class: ⩽50K.
Unusualness(Wracc): 0.080; significance: 114.745 confidence: 0.870; support: 0.592; coverage: 0.680.
This rule indicates that somebody who has less than medium educational level, without important capital-losses and capital-gains in the year, earns less or equal than 50K.

(2) IF age ⩽35.5 AND capital-loss ⩽1784.0 AND capital-gain ⩽7550.0 THEN Class: ⩽50K.
Unusualness(Wracc): 0.061; significance: 109.947 confidence: 0.893; support: 0.391; coverage: 0.438;
In this case, the rule shows that somebody younger than 35.5 years, with minimal capital-losses but medium capital-gains, earns less or equal than 50K.

(3) IF hours-per-week ⩽44.5 AND education-num ⩽ 12.0 AND capital-loss ⩽2215.5 AND capital-gain ⩽ 7550.0 THEN Class: ⩽50 k.
Unusualness(Wracc): 0.075; significance: 125.159 confidence: 0.886; support: 0.495; coverage: 0.559.
This rule represents the case of somebody with less than medium educational level, with medium capital-losses and capital-gains, working less than 44.5 h per week, who will earn ⩽50K.

(4) IF relationship = not-in-family AND capital-loss ⩽2215.5 AND capital-gain ⩽7550.0 THEN Class: ⩽50K.
Unusualness(Wracc): 0.043; significance: 100.877 confidence: 0.924; support: 0.234; coverage: 0.253.
This rule indicates that somebody who does not live in family, with medium capital-losses and capital-gains in the year, earns less or equal than 50K.

Some rules for the class *person who earn* >50K:

(1) IF marital-status ≠married-civ-spouse AND education-num >10.5 AND hours-per-week >22.5 AND age >28.0 THEN Class: >50K.
Unusualness(Wracc): 0.074; significance: 305.430 confidence: 0.721; support: 0.113; coverage: 0.157.
In this case, somebody who is not married, with medium-high educational level, working more than 22.5 h per week and older than 28 years, earns more than 50K.

(2) IF capital-gain >5100.0 AND sex ≠male THEN Class: >50K.
Unusualness(Wracc): 0.032; significance: 222.386 confidence: 0.984; support: 0.0429; coverage: 0.043.
This is a rule which indicates that a woman with more than medium capital-gain per year, earns more than 50K.

(3) IF relationship ≠husband AND education-num >10.0 AND age >28.0 THEN Class: >50K.
Unusualness(Wracc): 0.073; significance: 301.652 confidence: 0.720; support: 0.112; coverage: 0.155.
The rule shows the case of a woman who has no husband, with medium-high educational level, older than 28, earns more than 50K.

(4) IF education-num >10.0 AND relationship = husband AND hours-per-week >21.5 AND age >28.0 THEN Class: >50K.

Unusualness(Wracc): 0.072; significance: 301.822 confidence: 0.729; support: 0.109; coverage: 0.150. This rule shows the situation of the woman of the previous rule, when she has husband. In the same situation, with medium-high educational level, older than 28, but with husband, she has to work more than 21.5 hours per week to earn more than 50K.

As we can see, these rules offer interesting hidden information difficult to find by experts.

6. Concluding remarks

This paper addresses the scaling problem involved when CN2-SD is applied in large size data sets. To avoid the drawbacks introduced by data set size, we propose the use of instance selection previous the subgroup discovery task. An experimental study has been carried out to analyze the results offered with and without pre-processing in different size data sets.

The main conclusion reached is that instance selection algorithms can be applied as pre-processing stage allowing to maintain the quality of the subgroups discovered by the CN2-SD algorithm, increasing the interpretability of the subgroups and facing the scaling problem which appears in large size data sets.

As instance selection pre-process, we can stress the use of IB2 and EIS-CHC due to the high reduction achieved which is useful for us for applying CN2-SD in large size data sets. In particular, EIS-CHC shows very good results in combination with CN2-SD, and we can point out that this algorithm is a good choice in combination with CN2-SD.

Appendix A. Results for small size data sets

Tables A.1–A.7.

Table A.1
Instance selection and subgroup discovery in Iris Data Set

Algorithm	SIG	WRACC	SIZE	ANT	COB	CONF	SUP	COMP
CN2-SD	3.441	0.136	4.800	1.485	0.552	0.622	0.320	1.000
CNN + CN2-SD	3.161	0.086	4.700	1.420	0.316	0.609	0.191	0.893
Drop3 + CN2-SD	3.712	0.183	3.000	1.333	0.338	0.911	0.296	0.960
Ib2 + CN2-SD	2.597	0.085	4.200	1.328	0.374	0.570	0.210	0.940
Ib3 + CN2-SD	3.395	0.151	3.800	1.717	0.384	0.775	0.279	0.940
ICF + CN2-SD	3.469	0.155	3.000	1.567	0.476	0.742	0.313	0.993
EIS-CHC + CN2-SD	3.916	0.177	2.600	1.200	0.330	0.892	0.299	0.800

Table A.2
Instance selection and subgroup discovery in Glass Data Set

Algorithm	SIG	WRACC	SIZE	ANT	COB	CONF	SUP	COMP
CN2-SD	3.029	0.073	15.000	3.438	0.384	0.576	0.181	0.990
CNN + CN2-SD	2.606	0.054	11.700	3.841	0.303	0.482	0.141	0.974
Drop3 + CN2-SD	2.339	0.049	9.400	3.102	0.272	0.518	0.135	0.928
Ib2 + CN2-SD	2.637	0.050	10.800	3.993	0.336	0.470	0.141	0.967
Ib3 + CN2-SD	2.532	0.047	11.100	4.052	0.332	0.450	0.142	0.957
ICF + CN2-SD	2.362	0.047	9.500	2.935	0.263	0.473	0.126	0.974
EIS-CHC + CN2-SD	2.332	0.041	5.000	1.628	0.248	0.421	0.110	0.885

Table A.3
Instance selection and subgroup discovery in Led24Dig Data Set

Algorithm	SIG	WRACC	SIZE	ANT	COB	CONF	SUP	COMP
CN2-SD	3.992	0.052	30.000	3.080	0.186	0.431	0.071	1.000
CNN + CN2-SD	3.846	0.046	25.600	3.196	0.187	0.391	0.066	1.000
Drop3 + CN2-SD	2.898	0.038	10.200	3.165	0.112	0.381	0.050	0.800
Ib2 + CN2-SD	3.760	0.044	25.000	3.141	0.179	0.370	0.063	1.000
Ib3 + CN2-SD	3.878	0.049	25.000	3.596	0.191	0.391	0.069	0.995
ICF + CN2-SD	2.615	0.040	10.200	3.241	0.095	0.418	0.050	0.740
EIS-CHC + CN2-SD	3.288	0.025	7.600	1.722	0.271	0.213	0.051	0.910

Table A.4
Instance selection and subgroup discovery in Pima Data Set

Algorithm	SIG	WRACC	SIZE	ANT	COB	CONF	SUP	COMP
CN2-SD	1.655	0.053	31.600	2.375	0.442	0.652	0.281	1.000
CNN + CN2-SD	1.664	0.044	28.200	3.555	0.337	0.622	0.215	0.999
Drop3 + CN2-SD	1.891	0.063	12.600	3.072	0.446	0.680	0.303	0.997
Ib2 + CN2-SD	1.985	0.041	28.700	3.688	0.300	0.598	0.191	1.000
Ib3 + CN2-SD	1.835	0.040	25.900	3.245	0.296	0.590	0.187	0.974
ICF + CN2-SD	2.007	0.072	12.100	2.682	0.534	0.671	0.362	1.000
EIS-CHC + CN2-SD	2.316	0.074	2.000	1.000	0.500	0.683	0.341	1.000

Table A.5
Instance selection and subgroup discovery in Lymphography Data Set

Algorithm	SIG	WRACC	SIZE	ANT	COB	CONF	SUP	COMP
CN2-SD	0.673	0.010	23.500	3.647	0.373	0.348	0.137	1.000
CNN + CN2-SD	0.732	-0.002	21.700	3.736	0.252	0.276	0.085	0.929
Drop3 + CN2-SD	0.787	0.001	8.800	2.673	0.239	0.353	0.097	0.921
Ib2 + CN2-SD	0.738	0.004	21.600	3.751	0.317	0.310	0.111	0.986
Ib3 + CN2-SD	0.776	0.002	20.100	4.255	0.291	0.295	0.102	0.994
ICF + CN2-SD	0.757	0.005	8.500	2.512	0.267	0.374	0.113	0.967
EIS-CHC + CN2-SD	0.563	0.007	2.300	1.033	0.342	0.348	0.113	0.765

Table A.6
Instance selection and subgroup discovery in Contraceptive Data Set

Algorithm	SIG	WRACC	SIZE	ANT	COB	CONF	SUP	COMP
CN2-SD	2.266	0.032	56.000	3.402	0.392	0.439	0.166	1.000
CNN + CN2-SD	2.254	0.035	40.200	3.351	0.418	0.431	0.170	1.000
Drop3 + CN2-SD	2.596	0.042	24.300	3.691	0.393	0.468	0.175	1.000
Ib2 + CN2-SD	2.291	0.035	40.100	3.289	0.431	0.433	0.176	0.999
Ib3 + CN2-SD	2.628	0.069	39.400	3.373	0.433	0.448	0.178	1.000
ICF + CN2-SD	2.678	0.046	27.300	3.754	0.400	0.458	0.180	1.000
EIS-CHC + CN2-SD	2.412	0.034	4.200	1.730	0.332	0.488	0.156	0.932

Table A.7
Instance selection and subgroup discovery in Wine Data Set

Algorithm	SIG	WRACC	SIZE	ANT	COB	CONF	SUP	COMP
CN2-SD	4.471	0.188	6.800	2.502	0.323	0.941	0.305	0.989
CNN + CN2-SD	3.571	0.162	4.200	1.962	0.321	0.891	0.277	0.932
Drop3 + CN2-SD	3.384	0.157	3.400	1.433	0.326	0.848	0.270	0.922
Ib2 + CN2-SD	3.027	0.134	3.300	1.662	0.350	0.807	0.259	0.910
Ib3 + CN2-SD	3.802	0.171	4.200	1.922	0.348	0.883	0.297	0.949
ICF + CN2-SD	3.664	0.156	3.000	1.133	0.298	0.876	0.260	0.837
EIS-CHC + CN2-SD	3.515	0.146	3.000	1.000	0.302	0.860	0.247	0.787

References

Anderson, T. W. (1984). *An introduction to multivariate statistical analysis.* John Wiley and Sons.

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* (pp. 6–17).

Back, T., Fogel, D., & Michalewicz, Z. (1997). *Handbook of evolutionary computation*. Oxford University Press.

Berlanga, F., Del Jesus, M. J., Gonzalez, P., Herrera, F., & Mesonero, M. (2006). Multiobjective evolutionary induction of subgroup discovery fuzzy rules: A case study in marketing. *Lecture Notes in Computer Science, 4065*, 337–349.

Brightom, H., & Mellish, C. (2002). Advances in instance selection for instance based learning algorithms. *Data Mining and Knowledge Discovery, 6*, 153–172.

Cano, J.-R., Herrera, F., & Lozano, M. (2003). Using evolutionary computation as instance selection for data reduction in KDD: An experimental study. *IEEE Transactions on Evolutionary Computation, 7*(6), 561–575.

Cano, J.-R., Herrera, F., & Lozano, M. (2005). Stratification for scaling up evolutionary prototype selection. *Pattern Recognition Letters, 26*, 953–963.

Cano, J.-R., Herrera, F., & Lozano, M. (2007). Evolutionary stratified training set selection for extracting classification rules with trade off precision-interpretability. *Data and Knowledge Engineering, 60*(1), 90–108.

Chang, Y.-C., Lai, P.-C., & Lee, M.-T. (2007). An integrated approach for operational knowledge acquisition of refuse incinerators. *Expert Systems with Applications, 33*(2), 413–419.

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

Clark, P., & Niblett, T. (1991). Rule induction with CN2: Some recent improvements. In *Proceedings of the fifth European working session on learning* (pp. 151–163).

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research, 7*, 1–30.

Domingo, C., Gavaldá, R., & Watanabe, O. (2002). Adaptive sampling methods for scaling up knowledge discovery algorithms. *Data Mining and Knowledge Discovery, 6*(2), 131–152.

Eshelman, L. J. (1991). The CHC adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination. *Foundation of Genetic Algorithms, 1*, 265–283.

Friedman, M. (1940). A comparison of alternative tests of significance for the problem of m rankings. *Annals of Mathematical Statistics, 11*, 86–92.

Gamberger, D., & Lavrač, N. (2002). Generating actionable knowledge by expert-guided subgroup discovery. In *Proceedings of principles of data mining knowledge discovery* (pp. 163–174).

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

Gamberger, D., & Lavrač, N. (2002). Expert-guided subgroup discovery: Methodology and application. *Journal of Artificial Intelligence Research, 17*, 501–527.

Gamberger, D., Lavrač, N., & Wettschereck, D. (2002). Subgroup visualization: A method and application in population screening. In *Proceedings of the intelligent data analysis in medicine and pharmacology* (pp. 31–35).

Grochowski, M., & Jankowski, N. (2004). Comparison of instance selection algorithms II. Results and comments. In *Proceedings of the 7th international conference on artificial intelligence and soft computing* (pp. 580–585).

Han, J., & Kamber, M. (2000). *Data mining: Concepts and techniques*. Morgan Kaufman.

Hart, P. E. (1968). The condesed nearest neighbour rule. *IEEE Transactions on Information Theory, 18*(3), 431–433.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics, 6*, 65–70.

Huang, C.-L., & Wang, C.-J. (2006). A GA-based feature selection and parameters optimization for support vector machines. *Expert Systems with Application, 31*(2), 231–240.

Iman, R. L., & Davenport, J. M. (1980). Approximations of the critical region of the Friedman statistic. *Communications in Statistics*, 571–595.

Janssens, D., Brijs, T., Vanhoof, K., & Wets, G. (2006). Evaluating the perfomance of cost-based discretization versus entropy- and error-based discretization. *Computers and Operations Research, 33*, 3107–3123.

Kavšek, B., Lavrač, N., & Bullas, J. C. (2002). Rule induction for subgroup discovery: A case study in mining UK traffic accident data. In *Proceedings of the international multi-conference on information society* (pp. 127–130).

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

Kibbler, D., & Aha, D. W. (1987). Learning representative exemplars of concepts: An initial case of study. In *Proceedings of the 4th international workshop on machine learning* (pp. 24–30).

Kim, K.-J. (2006). Articial neural networks with evolutionary instance selection for financial forecasting. *Expert Systems with Applications, 30*(3), 519–526.

Klöesgen, W. (1996). Explora: A multipattern and multistrategy discovery assistant. In *Advances in knowledge discovery and data mining*. MIT Press, pp. 249–271.

Klöesgen, W., & Michael, M. (2002). Census data mining – An application. In *Proceedings of the 6th European conference on principles of data mining, knowledge discovery* (pp. 65–79).

Kuncheva, L. (1995). Editing for the k-nearest neighbors rule by a genetic algorithm. *Pattern Recognition Letters, 16*, 809–814.

Lavrač, N., Flach, P., Kavšek, B., & Todorovski, L. (2002). Adapting classification rule induction to subgroup discovery. In *Proceedings of the IEEE international conference on data mining* (pp. 266–273).

Lavrač, N., Cestnik, B., Gamberger, D., & Flach, P. (2004). Decision support through subgroup discovery: Three case studies and the lessons learned. *Machine Learning, 57*, 115–143.

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

Lavrač, N. (2005). Subgroup discovery techniques and applications. In *Proceedings of the 19th Pacific-Asia conference advances in knowledge discovery* (pp. 2–14).

Liu, H., & Motoda, H. (Eds.). (2001). *Instance selection and construction for data mining*. Kluwer Academic Publishers.

Liu, H., Hussain, F., Tan, C. L., & Dash, M. (2002). Discretization: An enabling technique. *Data Mining and Knowledge Discovery, 6*(4), 393–423.

Liu, H., & Motoda, H. (2002). On issues of instance selection. *Data Mining and Knowledge Discovery, 6*, 114–130.

Nakada, T., & Kunifuji, S. (2003). Subgroup discovery among personal homepages. In *Proceedings of the discovery science* (pp. 385–392).

Newman, D. J., Hettich, S., Blake, C. L. & Merz, C. J. (1998). UCI Repository of machine learning databases, http://www.ics.uci.edu/~mlearn/MLRepository.html, University of California, Irvine, Department of Information and Computer Sciences.

Polat, K., & Gunes, S. (2007). Medical decision support system based on artificial immune recognition immune system (AIRS), fuzzy weighted pre-processing and feature selection. *Expert Systems with Applications, 33*(2), 484–490.

Provost, F., & Kolluri, V. (1999). A survey of methods for scaling up inductive algorithms. *Data Mining and Knowledge Discovery, 3*(2), 131–169.

Riquelme, J. C., Aguilar, J. S., & Toro, M. (2003). Finding representative patterns with ordered projections. *Pattern Recognition, 36*, 1009–1018.

Sánchez, J. S. (2004). High training set size reduction by space partitioning and prototype abstraction. *Pattern Recognition, 37*, 1561–1564.

Sebban, M., Nock, R., Chauchat, J. H., & Rakotomalala, R. (2000). Impact of learning set quality and size on decision tree perfomances. *International Journal of Computers, Systems and Signals, 1*(1), 85–105.

Shang, W., Huang, H., Zhu, H., Lin, Y., Qu, Y., & Wang, Z. (2006). A novel feature selection algorithm for text categorization. *Expert Systems with Applications, 33*(1), 1–5.

Sheskin, D. J. (2000). *Handbook of parametric and nonparametric statistical procedures*. CRC Press.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics, 1*, 80–83.

Wilson, D. L. (1972). Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man and Cybernetics, 2*(3), 408–421.

Wilson, D. R., & Martinez, T. R. (1997). Instance pruning techniques. In *Proceedings of the 14th international conference* (pp. 403–411).

Wilson, D. R., & Martinez, T. R. (2000). Reduction techniques for instance-based learning algorithms. *Machine Learning, 38*, 257–268.

Wrobel, S. (1997). An algorithm for multi-relational discovery of subgroups. In *Proceedings of the 1st European conference on principles of data mining and knowledge discovery* (pp. 78–87).

Yen, S.-J., & Lee, Y.-S. (2006). An efficient data mining approach for discovering interesting knowledge from customer transactions. *Expert Systems with Applications, 30*(4), 650–657.

Zar, J. H. (1999). *Biostatistical analysis*. Prentice Hall.

Železný, F., & Lavrač, N. (2006). Propositionalization-based relational subgroup discovery with RSD. *Machine Learning, 62*, 33–63.