Influence of Missing Values Handling on Classification Rules Evolved from Medical Data

Diana Lungeanu[1], Daniela Zaharie[2], and Flavia Zamfirache[2]

[1]Department of Medical Informatics, University of Medicine and Pharmacy
2 E. Murgu, Sq., Timişoara, Romania
[2]Department of Computer Science, West University of Timişoara,
4 V. Pârvan, blvd., Timişoara, Romania
dlungeanu@umft.ro,dzaharie@info.uvt.ro,zflavia@info.uvt.ro

Abstract. The aim of this work is to analyze the influence of several methods to handle the missing values when searching for classification rules in medical data. The rule mining process is based on a multi-objective evolutionary algorithm and we analyzed the impact on the evolved rules' quality of a simple imputation method and of two strategies which deal with missing values during the mining process. The proposed methods interfere with the evolutionary process by influencing the evaluation of the criteria used to measure the rules accuracy and interestingness. Preliminary tests, conducted on medical data, suggest that there are no statistically significant differences between the analyzed methods. However the methods dealing with the missing values during the mining process leads to slightly better rules than the pre-processing strategy based on median imputation.

1 Introduction

Medical data mining is specifically aimed at providing reliable knowledge in a comprehensible form. In the particular case of classification problems, black-box classifiers (e.g. neural networks) do not always satisfy the comprehensibility requirements. Therefore it can be helpful to extract rules either from trained classifiers or directly from data. The work presented in this paper followed the latter approach.

Discovering rules in data is a search process usually guided by several measures based on which the potential quality and usefulness of rules are evaluated (accuracy, interestingness and comprehensibility measures). These measures are usually conflicting, i.e. an accurate rule is not necessarily interesting or easy to read, thus the searching process has to be multicriterial. Evolutionary algorithms (EAs) proved to be valuable instruments in discovering rules from data [1], [4]. Approaches based on multi-objective evolutionary algorithms (MOEAs) have also been proposed [3].

On the other hand, the medical data encountered in practice usually suffer from a significant number of missing values (MVs). Although every effort should be made towards avoiding sparse data collection of medical data (as they may

lead to biased conclusions), almost always there are some missing data and they should be handled with care.

The most radical approach to deal with MVs is to simply eliminate the observations with any missing variable values (i.e. the incomplete cases) from the analysis. Although a very simple solution, this may result in severe reduction of available data and lost of potentially valuable information in the incomplete cases [6]. Another approach can be to divide the database with missing values into several valid databases (i.e. for a rule, a valid database must not have any missing values on attributes involved in the rule) and redefine the quality measures for each such database [8]. Other strategies are focused on pre-processing the data by imputation (i.e. substituting a value for each missing piece of data), usually employing the most common value, or another value reflecting the central tendency for that attribute [9].

Many efforts have been made for finding reliable solutions for mining rules from incomplete data, but they were mostly focused on association rules [6], [7],[8]. Being involved in a study targeted at discovering classification rules for preterm-birth prediction, we faced the problem of incomplete data. Moreover, some attributes had up to 90% absent values, while still could not be completely ignored in the classification sought. This created serious difficulties in the mining process and motivated us to try to find a practical solution. To the best of our knowledge, the impact of the different methods of handling the MVs in mining classification rules by evolutionary algorithms has not been analyzed and we have not found any reported results on the robustness of evolutionary rule mining. Taking all these into account, we undertook such an analysis for medical data.

2 Evolutionary Rule Mining

The core of our mining approach is a multi-objective evolutionary algorithm (MOEA) which evolves a population of classification IF-THEN rules having in the antecedent part a conjunction of conditions satisfied by attributes. The evolutionary process aims to identify rules corresponding to one class thus the problem should be interpreted as a binary classification one. For instance when the aim is to evolve rules corresponding to the class of patients having diabetes a candidate rule could be *IF Plasma glucose concentration >123 AND Body mass index >23 THEN diabetes class.* Since the class to be described by rules is fixed, only the antecedent part of the rule should be discovered and therefore encoded in the population. The general structure of the MOEA is described in Algorithm 1 and the particularities of the rules encoding and of the corresponding evolutionary operators are described in the following.

Rules encoding. Each element of the population encodes the antecedent part of a rule and it consists of a list of components corresponding to all attributes in the data set. Each component consists of three fields: *(presence flag, operator, value)*. The *presence flag* is a binary value specifying whether the corresponding attribute is involved in the rule. The *operator* allows specifying the condition the attribute should satisfy. We used two possible operators for each type of

Algorithm 1 Generic MOEA for rules extraction

```
Initialize a population of m rules
Evaluate the population
while "the maximal number of generations is not reached" do
    Generate m new rules by crossover
    Apply mutation to rules obtained by crossover
    Evaluate the new elements
    Join the old and the new populations
    Select the best m rules from the joined population
end while
```

attributes. In the case of numerical attributes the possible operators are $\leq$ (coded by 0) and $>$ (coded by 1). For the categorical attributes the operators are $=$ (coded by 1) and $\neq$ (coded by 0). The *value field* contains the value associated to the attribute.

Crossover and mutation. During each generation, a new population is constructed by crossover and mutation from the elements of the current population. By crossover, a new rule is constructed starting from two randomly selected rules. The crossover procedure can be described as following: (i) if the attribute is absent from both parent rules, it will be absent from the generated rule, as well; (ii) if the attribute is present in only one parent rule, its operator and value field are transferred to the new rule; (iii) if the attribute is present in both rules and satisfies the same type of condition (the operator field has the same value in both rules) then it is transferred to the new rule: for numerical attributes, the new value is the average of the values corresponding to the parent rules; for nominal attributes, one of the parents values is just randomly taken; (iv) if the attribute is present in both rules but it satisfies different conditions, then the triplet to be transferred into the new rule is taken from the parent rule which is better with respect to one of the evaluation criteria (in our experiments we used the first criterion specified by the user).

The mutation has the role of modifying the rules obtained by crossover. For each attribute, mutation is applied with a given probability (e.g. $p_m = 0.1$) and it can affect one of the fields (i.e. presence flag, operator or value) and only one at each mutation step. By switching the presence flag, some attributes can be inserted or removed from the rule, thus leading to either a more general or a more specific rule. By changing the operator field, one changes the condition the attribute should satisfy. The mutation of the value field consists in choosing a new value based on a uniform selection from the range of values corresponding to the attribute.

Rules evaluation. The generated rules are evaluated at each generation with respect to several measures which determine their fitness. In our analysis we used quality measures specific to classification problems (e.g. accuracy, specificity and sensitivity) and measures quantifying the rule interestingness (e.g. uncovered negative [5] and lift). Let us consider the case of searching for rules corresponding

to a class C in a data set X ($X = C \cup \overline{C}$). All above mentioned measures can be computed for a rule R by using the following probabilities: $P(R)$ (probability to satisfy the antecedent of the rule R), $P(C)$ (probability to belong to class C), $P(R, C)$ (probability to satisfy the rule R and to belong to the class C) and the probabilities corresponding to the complementary events ($P(\overline{R})$, $P(\overline{C})$, $P(\overline{R}, \overline{C})$). Based on these probabilities the measures used in our analysis can be defined as follows:

$$\begin{array}{ll} acc = P(R,C) + P(\overline{R},\overline{C}) & lift = P(R,C)/(P(R)P(C)) \\ spec = P(\overline{R},\overline{C})/P(\overline{C}) & UN = P(\overline{R},\overline{C}) \\ sens = P(R,C)/P(C) & \end{array} \quad (1)$$

A rule is better as all these values are larger. If there are no missing values in the data, the probabilities involved in Eq.(1) are estimated by using relative frequencies. When there are missing values in the data they have to be taken into account, thus different ways for computing the frequencies can be used. This aspect will be discussed in the next section.

Selection. Since several quality criteria are used to evaluate the rules there is no just one best rule but several which are reciprocally non-dominated. A rule is considered as non-dominated, with respect to rules in a given set, if no other rule in that set is better with respect to all criteria. The set of all non-dominated rules form the so-called Pareto optimal set and the corresponding values of the evaluation criteria define the Pareto front. The aim of any MOEA is to find a good approximation of the Pareto front and implicitly of the Pareto optimal set. Our selection strategy is similar to that used in NSGA-II [2], meaning that the elements in the joined population (parents and offspring) are ranked based on the non-domination relationship.

The elements that are non-dominated with respect to all elements in the joined population belong to the first nondomination set and have the rank 1. Subsequently, the nondominated elements in the population obtained by ignoring the elements of rank 1 belong to the second nondomination set and so on. The m elements corresponding to the new generation are selected from the $2m$ ranked elements in their ranks increasing order. For stimulating the diversity of the resulting approximation of the Pareto front, a crowding distance is used as a second selection criterion: from two elements having the same rank, the one with a larger crowding distance (suggesting that it belongs to a less crowded region) is selected. A particular characteristic of our approach is related to the crowding distance between rules. We used two types of distances, one expressing the structural difference between rules and another expressing the difference between the data subsets covered by the rules. In the case of two rules R and R' encoded by lists of n tuples ($t = (p, o, v)$) the structural distance is defined as follows:

$$d_S(R,R') = \frac{\sum_{j=1}^{n} d_j(R,R')}{n}, \qquad d_j(R,R') = \begin{cases} 0 & \text{if } p_j = p'_j, \quad o_j = o'_j \\ 1 & \text{if } p_j = p'_j, \quad o_j \neq o'_j \\ 2 & \text{if } p_j \neq p'_j \end{cases}, \quad (2)$$

where p_j denotes the presence flag and o_j denotes the operator corresponding to jth attribute. Thus two rules are considered to be identical from a structural point of view if they contain the same attributes and the terms have identical associated operators. The distance related with the rule coverage is defined as the cardinal of the subset of data which are either covered by the first rule but are not covered by the second rule, or are covered by the second rule and are not covered by the first one. Thus the cover-based distance is: $d_C(R, R') = \text{card}(C(R)\Delta C(R'))$, where $C(R)$ is the set of data instances which satisfy the rule (are covered by the rule) and Δ denotes the symmetrical difference between two sets. These distances are also used to filter the final results in order to eliminate some of the rules which are to close to each other.

3 Handling Missing Values

When we evaluate a rule with respect to a set of incomplete data we have to decide how the missing values will be handled. A first variant would be to pre-process the data and to impute the missing values using one of the well-known imputation methods [7],[9]. The variant we analyze is based on the idea of dealing with missing values during the rule mining process. The step when we have to take into account the existence of missing values is while checking whether or not a record with MVs matches a given rule. To be more specific let us consider the case of a rule containing k terms, $R = (T_1, \ldots, T_k)$, where each term T_q describes a property of an attribute i_q (e.g. $a_{i_q} \in V_q^R$). For instance for a rule R=IF$(a_2 = 5)$AND$(a_3 \leq 3.5)$AND$(a_5 > 4)$ THEN C, which involves three attributes ($i_1 = 2$, $i_2 = 3$, $i_3 = 5$), the corresponding sets of values are: $V_1^R = \{2\}$, $V_2^R = [inf_{i_2}, 3.5]$, $V_3^R = (4, sup_{i_3}]$ (inf_{i_2} is the lower bound of the possible values for attribute i_2 and sup_{i_3} is the upper bound of the possible values for attribute i_3). Using these notations we can define the probabilities involved in Eq. (1).

Let us first start with the case of complete data (no missing values). In such a situation for each rule $R = (T_1, \ldots, T_k)$ and a data record $x = (x_1, \ldots, x_n)$ one can define the matching value:

$$\mu(x, R) = \begin{cases} 1, & x_{i_q} \in V_q^R, q = \overline{1,k} \\ 0, & \text{there exists } q \text{ with } x_{i_q} \notin V_q^R \end{cases}. \tag{3}$$

Using this matching value one can compute a cover measure expressing the number of elements in a subset $S \subseteq X$ of data which match the rule:

$$cover(R, S) = \sum_{x \in S} \mu(x, R). \tag{4}$$

Similarly, one can define $cover(\overline{R}, S) = \sum_{x \in S}(1 - \mu(x, R))$. Thus the probabilities used to evaluate the rules can be estimated using the following frequencies:

$$\begin{aligned} P(R) &= cover(R, X)/cardX & P(R, C) &= cover(R, C)/cardX \\ P(\overline{R}) &= cover(\overline{R}, X)/cardX & P(\overline{R}, \overline{C}) &= cover(\overline{R}, \overline{C})/cardX. \end{aligned} \tag{5}$$

Let us turn now to the case of incomplete data. When evaluating a rule R, the missing values can interfere with the evaluation process only when they correspond to attributes involved in the rule. Therefore we have to decide on how the computation of the above probabilities should be modified in such a situation. We analyze three variants.

Variant 1. The simplest variant is to consider that a data x which has a missing value on at least one attribute involved in the rule does not match the rule (even if all other terms in the rule are satisfied). Thus the only change is in the computation of $\mu(x, R)$ which becomes:

$$\mu(x,R) = \begin{cases} 1, & x_{i_q} \in V_q^R, q = \overline{1,k} \\ 0, & \text{there exists } q \text{ such that } x_{i_q} \notin V_q^R \text{ or } x_{i_q} \text{ is missing} \end{cases} . \quad (6)$$

All the other equations remain unchanged. The main impact of this change is the fact that the cover measure of rules involving attributes with missing values will be smaller. Since the denominator used in Eqs. (5) is unchanged the corresponding probabilities will be smaller. Therefore, this first variant penalizes rules which involve incomplete attributes, leading to a smaller chance for survival during the evolutionary process.

Variant 2. In this variant we try to limit the penalization of rules involving incomplete attributes. Therefore the probabilities given in Eqs. (5) will be computed not with respect to X but with respect to $S(R) \subseteq X$ which contains only data having specified all attributes involved in R (these data are considered valid with respect to this rule). This means replacing X with $S(R)$ in Eqs.(5). By this change the rules involving incomplete attributes are not so drastically penalized as in the previous variant. This approach is similar to that used in [8] to compute the support in the case of extracting association rules from incomplete data.

Variant 3. Another variant is to not ignore the incomplete records but to penalize them by using a non-crisp matching value. Instead, of the 0 match value for a missing value, each missing attribute value will be assigned a probability to satisfy the corresponding term of the rule. The matching value, $\mu(x, R)$, will be in this case defined as follows:

$$\mu(x,R) = \prod_{q=1}^{k} \nu(x, T_q), \qquad \nu(x, T_q) = \begin{cases} 1, & x_{i_q} \in V_q^R \\ 0, & x_{i_q} \notin V_q^R \\ \pi(V_q^R), & x_{i_q} \text{ is missing} \end{cases} . \quad (7)$$

The probability $\pi(V_q^R)$ corresponding to a term T_q is defined depending on the attribute type and term structure. If the attribute a_{i_q} is discrete then V_q^R is a finite set. If the term T_q is of equality type $(a_{i_q} = v_q)$ then $\pi(V_q^R) = 1/card(V_q^R)$ and if it is of inequality type $(a_{i_q} \neq v_q)$ then $\pi(V_q^R) = (card(V_q^R) - 1)/card(V_q^R)$. In the case of continuous numeric attributes their values usually belong to a bounded interval (e.g. $[inf_{i_q}, sup_{i_q}]$). If the term in the rule is based on a "less than" operator $(a_{i_q} \leq v_q)$ then $\pi(V_q^R) = (v_q - inf_{i_q})/(sup_{i_q} - inf_{i_q})$ and if it is based on a "greater than" operator $(a_{i_q} > v_q)$ then $\pi(V_q^R) = (sup_{i_q} - v_q)/(sup_{i_q} - inf_{i_q})$.

This is just a variant of estimating the probability that a missing value would satisfy the rule term, based on the simplifying assumption that the values of the attributes are uniformly distributed on their range. By using other distribution models for the attributes values, different matching values would be obtained. The cover measure and the corresponding probabilities are further computed as in Eqs. (4) and (5). The main difference between the first variant and the current one is the fact that in the first case $cover(R, S) \in \{1, \ldots, cardS\}$ while in the second one $cover(R, S) \in [1, cardS]$.

4 A Comparative Analysis for Medical Data

In order to analyze the impact of different methods of handling missing values on the quality of evolved rules we conducted some experiments on two datasets from the UCI Machine Learning repository (http://mlearn.ics.uci.edu/ MLRepository.html): Pima indians diabetes and Wisconsin breast cancer (1991) datasets.

Data preparation. The data were prepared for tests by randomly eliminating $mv\%$ values of attributes ($mv \in \{10, 20, 30\}$ and it represented a percent of the number of total number of attributes in the dataset). For cross-validation, the data were randomly split in five folds. The missing values were introduced only in the data used for training, the validation sets being with the original values of attributes.
Methods for handling missing values. We analyzed three methods: the first one (denoted as Method 1) corresponds to variant 2 described in the previous section; the second one (Method 2) is in fact the variant with non-crisp matching values; and Method 3 is based on a simple imputation strategy (the data are pre-processed such that the missing values are replaced with the median of all existing values for the corresponding attribute).
The classification rules mining. For all datasets we considered the problem of evolving classification rules corresponding to one class (e.g. the malignant class in breast cancer dataset). We used the multi-objective evolutionary algorithm described in Section 2 in order to evolve rules which simultaneously maximize two criteria: the product between specificity and sensitivity ($spec * sens$) and the uncovered negative measure (UN). This last one was used since results presented in [5] suggest that it is an adequate interestingness measure for medical data. The parameters of the algorithm were set as follows. The population size was set to 50, the mutation probability was set to 0.1 and the number of generations to 100. This rather small number of generations was used since our primary goal was to compare the methods for treating the missing values and not to necessarily obtain high accurate rules. After 100 generations the algorithm provides a set of rules which are reciprocally non-dominated with respect the chosen criteria. This set of rules can be interpreted as a classification ruleset. Therefore in the testing step the matching between each test data and each rule in the set is checked and if the data matches with at least one rule then is counted a match. This number of matches between the test data and the set of rules is further used to compute quality measures of the ruleset (accuracy, specificity, sensitivity, lift,

uncovered negative). In order to allow the comparison between different runs, at each run the evolutionary algorithm starts from the same initial population. The same operators and control parameters were used in all cases. Thus the differences remarked in the final results are entirely determined by the impact of the different methods for handling missing values.

Results. The experiments illustrated that using different methods for handling the missing values different sets of rules are obtained. Table 1 presents the rules with the highest value of $spec * sens$ from the sets evolved by the analyzed methods. In the case of Method 1 were evolved 18 rules, in the second case were evolved 9 and in the last one 19 rules.

Table 1. Examples of rules extracted from the Pima indians diabetes data set with 10% of missing values

Method	Rule with highest value of the product $spec * sens$
1	IF (Plasma glucose concentration >110) AND (2-h serum insulin<532.4) AND (BMI>28.73) AND (Diabetes pedigree function $\neq$0)AND(Age>28) THEN diabetes class ($acc = 0.75, spec = 0.86, sens = 0.55, UN = 0.55, lift = 9.72$)
2	IF (Plasma glucose concentration > 108.47) AND (BMI>26.38) THEN diabetes class ($acc = 0.68, spec = 0.67, sens = 0.72, UN = 0.43, lift = 1.54$)
3	IF (Plasma glucose concentration >110.32) AND (2-h serum insulin<533.98) AND (BMI> 24) AND (age > 24) THEN diabetes class ($acc = 0.72, spec = 0.77, sens = 0.64, UN = 0.5, lift = 1.72$)

On the other hand, the results in Tables 2 suggest that the differences in the quality of the obtained rulesets are not statistically significant. Slightly better results were obtained for methods which adjust the computation of the quality measures in order to deal with MVs (Method 1 and Method 2).

This is especially remarked in the case of breast cancer dataset when the method based on non-crisp matching values (Method 2) leads to results comparable with those obtained for complete data.

We further conducted a similar analysis on a set of obstetrical data containing information about births collected during one year (2006) at one regional hospital of obstetrics and gynaecology. The set of data contains 2686 records corresponding to two classes: the class of pre-term births (370 records, representing 13.77%) and the class of on-term births (2316 records, representing 86.23%). Each record contains 63 attributes corresponding to different characteristics of mothers and new-born children. The overall percentage of missing values is 23% but they are non-uniformly distributed over attributes. There are attributes with a very small percent of missing values (e.g. the value of the mother age is missing in only two cases, representing 0.07%) and attributes with a high percent of missing values (e.g. the value of the biparietal diameter which is missing in 90% of cases). Our aim was to evolve rules corresponding to the pre-term births class. The large

Table 2. Results for two data sets from UCI Machine Learning repository

	mv	Method 1			Method 2			Method 3		
	0%	10%	20%	30%	10%	20%	30%	10%	20%	30%
	Pima indians diabetes data set									
acc	0.72±	0.72±	0.66±	0.67±	0.71±	0.67±	0.67±	0.71±	0.71±	0.72±
	0.02	0.01	0.03	0.05	0.04	0.07	0.07	0.05	0.05	0.04
spec	0.74±	0.74±	0.64±	0.68±	0.72±	0.66±	0.63±	0.71±	0.7±	0.73±
	0.05	0.04	0.12	0.08	0.08	0.18	0.19	0.07	0.1	0.08
sens	0.68±	0.68±	0.72±	0.64±	0.69±	0.68±	0.76±	0.72±	0.72±	0.71±
	0.09	0.08	0.2	0.09	0.09	0.14	0.16	0.04	0.06	0.04
UN	0.48±	0.48±	0.41±	0.44±	0.47±	0.43±	0.41±	0.46±	0.45±	0.47±
	0.03	0.03	0.08	0.05	0.05	0.11	0.12	0.04	0.06	0.05
lift	1.68±	1.70±	1.51±	1.52±	1.66±	1.58±	1.58±	1.66±	1.65±	1.7±
	0.09	0.06	0.1	0.18	0.19	0.26	0.2	0.19	0.19	0.17
	Breast cancer data set									
acc	0.9±	0.89±	0.88±	0.87±	0.92±	0.91±	0.92±	0.85±	0.86±	0.88±
	0.06	0.04	0.05	0.06	0.04	0.06	0.03	0.04	0.02	0.05
spec	0.89±	0.88±	0.83±	0.85±	0.93±	0.9±	0.92±	0.8±	0.79±	0.84±
	0.07	0.06	0.07	0.11	0.04	0.06	0.04	0.04	0.03	0.07
sens	0.94±	0.92±	0.96±	0.91±	0.92±	0.9±	0.91±	0.95±	1±	0.95±
	0.04	0.05	0.03	0.13	0.07	0.09	0.06	0.09	0	0.07
UN	0.64±	0.58±	0.54±	0.55±	0.6±	0.59±	0.6±	0.52±	0.51±	0.55±
	0.15	0.04	0.04	0.07	0.03	0.04	0.02	0.02	0.02	0.04
lift	3.98±	2.39±	2.2±	2.34±	2.57±	2.47±	2.54±	2.1±	2.09±	2.25±
	3.64	0.25	0.23	0.35	0.21	0.31	0.19	0.16	0.09	0.21

number of attributes leads to a very large search space, creating difficulties for the evolutionary algorithm to discover high quality rules. Therefore we selected various subsets of attributes to be involved in the rules. The results presented in Table 3 were obtained when using as possible antecedents in the classification rules the information about previous pregnancies, miscarriages and abortions and about the fundal height. This last attribute has 16% of missing values. We used the product between the specificity and sensitivity as unique optimization criterion. The optimization being with a single criterion each run led to only one rule. All obtained rules contain a term corresponding to "fundus uterus" height attribute (e.g. "fundus uterus" height$\leq$ 29) even if this attribute has missing values while the other ones do not contain missing values. This can be explained by the fact that the fundal height attribute has a predictive value for the preterm birth. On the other hand, as Table 3 suggests the results obtained by the three analyzed methods are not significantly different. However the first method leads to slightly better results than the other two.

5 Conclusions

We proposed several methods to deal with missing values when evolving classification rules from data. All of them take into account the absence of some

Table 3. Results for obstetrical dataset

	acc	spec	sens	UN	lift
Method 1	0.70±0.09	0.73±0.13	0.52±0.17	0.58±0.12	1.92±0.63
Method 2	0.64±0.08	0.64±0.12	0.59±0.18	0.55±0.1	1.57±0.34
Method 3	0.64±0.05	0.66±0.07	0.54±0.11	0.57±0.06	1.54±0.29

attributes' values by modifying the estimation of probabilities involved in the evaluation of rules quality. A comparative analysis involving two of the proposed methods and a simple imputation strategy were conducted for two datasets from UCI repository and a database containing obstetrical data. Even if no statistically significant differences were remarked, the methods which interfere with the evaluation of the rule's quality led to slightly better results than an imputation method based on the median.

Acknowledgment. This work is supported by the grant 99-II CEEX 03 - INFOSOC 4091/31.07.2006 from the Romanian Ministry of Education and Research (http://www.maternqual.ro).

References

1. Araujo, D.L.A., Lopes, H.S., and Freitas., A.A.: Rule Discovery With A Parallel Genetic Algorithm. Proc. 2000 Genetic and Evolutionary Computation, Las Vegas, NV, USA, pp. 89-92 (2000)
2. Deb, K., Agrawal S., Pratab, A., and Meyarivan, T.: A Fast Elitist Non-Dominated Sorting Genetic Algorithm For Multi-Objective Optimization: NSGA-II. In: M. Schoenauer, et al. (eds), Proceedings of PPSN, Lecture Notes in Computer Science, vol 1917, pp. 849-858 (2000)
3. Dehuri, S, Patnaik, S., Ghosh, A., and Mall., R.: Application of Elitist Multi-Objective Genetic Algorithm For Classification Rule Generation. Applied Soft Computing 8(1), 477-487 (2008)
4. Fidelis, M.V., Lopes., H.S., and Freitas, A.A.: Discovering Comprehensible Classification Rules with a Genetic Algorithm, Proceedings of CEC2000, La Jolla, CA, USA, pp. 805-810 (2000)
5. Ohsaki, M., Abe, H., Tsumoto, S., Yokoi, H., and Yamaguchi., T.: Proposal of Medical KDD Support User Interface Utilizing Rule Interestingness Measures. In: Proceedings of ICDMW'06, IEEE, pp. 759-764 (2006)
6. Ragel, A., Cremilleux, B.: Treatment of Missing Values for Association Rules. In: Proc. of the Second Pacific-Asia Conference on Knowledge Discovering and Data Mining (PAKDD-98), LNAI vol. 1394, pp. 258-270 (1998)
7. Ragel, A., Cremilleux, B.: MVC - A Preprocessing Method to Deal with Missing Values. Knowledge-Based Systems 12(5-6), pp. 285-291 (1999)
8. Shintani, T.: Mining Association Rules from Data with Missing Values by Database Partitioning and Merging. In: Proc. of the ICIS-COMSAR'06, IEEE Computer Science, pp.193-200 (2006)
9. Yuan, Y.: Multiple Imputation for Missing Values: Concepts and New Development SAS Institute Inc., SUGI Proceedings, paper P267-25 (2000), http://www2.sas.com/proceedings/sugi25/25/st/25p267.pdf