An Integer Programming Approach to Inductive Learning
Using Genetic Algorithm

Janusz Kacprzyk and Grazyna Szkatula
Systems Research Institute, Polish Academy of Sciences
ul. Newelska 6, 01-447 Warsaw, Poland
E-mail: kacprzyk@ibspan.waw.pl

Abstract - We propose an improved inductive learning method
to derive classification rules that correctly describe most of the
examples belonging to a class and do not describe most of the
examples not belonging to this class. The problem is represented
as a modification of the set covering problems solved by a
genetic algorithm. The results are very encouraging.

I. INTRODUCTION

Machine learning from examples is a process of
inferring a classification rule (concept description) of a
class from positive and negative examples.
In practice, mainly due to imperfect data, the
requirements to be satisfied by learning procedures are:
• a partial completeness, i.e. that the classification
rule must correctly describe (have the same attribute
values), say, most of the positive examples,
• a partial consistency, i.e. that the classification rule
must describe, say, almost none of the negative examples,
• convergence, i.e. the classification rule must be
derived in a finite number of steps,
• the classification rule of minimal "length" is to be
found, e.g. with the minimum number of attributes
(or, more generally being "simple").

Examples are described (cf. Michalski [22]) by a set
of K "attribute - value" pairs

\[ e = \land_{j=1}^{K} [a_j \# v_j] \]

where \( a_j \) denotes attribute \( j \) with value \( v_j \) and \# is a
relation as, e.g., =, <, >, =\geq, etc. For instance, for the
attributes: height, color_of_hair, color_of_eyes, we can
describe the look of a person as [height = "high"] \land
[color_of_hair = "blond"] \land [color_of_eyes = "blue"].

We propose a modified inductive learning procedure
based on Michalski’s [22] star-type methodology. The method is based on elements of the authors’ previous
work [cf. Kacprzyk and Szkatula [15 – 20]. A pre-
processing of data (examples) is first performed based
on an analysis of how frequent the values of the
particular attributes occur in the examples. These
frequencies are used to derive weights associated with
those values, and the problem is a modification of the
set covering problems, and solved by a modification of
a genetic algorithm (IP2_GA).

II. PROBLEM FORMULATION OF INDUCTIVE
LEARNING FROM EXAMPLES

We have finite sets of examples \( U \) and attributes
\( A = \{a_1, ..., a_K\} \). \( V_{a_j} = \{v_{a_j}^{a_j}, ..., v_{a_j}^{a_j}\} \) is a domain \( a_j \),
\( j = 1, ..., K \), \( V = \bigcup_{j=1}^{K} V_{a_j} \). \( f : U \times A \rightarrow V \) is a function
such that \( f(e, a_j) \in V_{a_j} \) for \( \forall a_j \in A \), \( \forall e \in U \), called
an informational function. Each \( e \in U \) is described by
K attributes, \( A = \{a_j, ..., a_K\} \), and is represented by

\[ e = \land_{j=1}^{K} [a_j = v_{i,j}] \] (1)

where \( v_{i,j} = f(e, a_j) \in V_{a_j} \) denotes the j-th attribute \( a_j \)
taking on a value \( v_{i,j} \) for example \( e \).

An example \( e \) in (1) is composed of K "attribute-
value" pairs, denoted \( s_j = [a_j = v_{i,j}] \) (selectors). The
conjunctive of \( I \subseteq K \) "attribute-value" pairs, i.e.

\[ C^I = \land_{j \in I} s_j = \land_{j \in I} [a_j = v_{i,j}] = \]
\[ \land_{j_1, ..., j_l} [a_{j_1} = v_{i,j_1}^{a_j}] \land ... \land [a_{j_l} = v_{i,j_l}^{a_j}] \] (2)

where \( I = \{j_1, j_2, ..., j_l\} \subseteq \{1, ..., K\} \) is called a complex.

Suppose we have example \( e \) [cf. (1)] and consider a
complex [cf. (2)] \( C^I = [a_{j_1} = v_{i,j_1}^{a_j}] \land ... \land [a_{j_l} = v_{i,j_l}^{a_j}] \)
that corresponds to the set of indices \( I = \{j_1, ..., j_l\} \subseteq \{1, ..., K\} \). The set of indices \( \{j_1, ..., j_l\} \) is
clearly equivalent to a vector \( x = [x_j]^T, \ j = 1, ..., K \),
such that \( x_j = 1 \) if a selector \( s_j = [a_j = v_{i,j}^{a_j}] \) occurs in
the complex \( C^I \), and 0 otherwise. For instance, for \( K = \)
3 and e = [height = "high"] \land [color_of_hair = "blond"] \land [color_of_eyes = "blue"], vector [0,1,0]^T is equivalent to the complex [color_of_hair = "blond"]; the complex
[height = "high"] is equivalent to the vector [1,0,0]^T.

Complex $C^l$ covers example $e$ if all conditions on attributes given as selectors are covered by (equal to) the values of the respective attributes in $e$, i.e. $f(C^l,a_j) = f(e,a_j), \forall j \in I$.

We assume that $a_d$ is a decision attribute and
$V_{a_d} = \{v_{a_d}^1, \ldots, v_{a_d}^{N_{a_d}}\}$ is a domain of $a_d$. Each example $e \in U$ is described by a set of attributes $\{a_1, a_2, \ldots, a_K\} \cup \{a_d\}$. Observe that attribute $a_d$ determines a partition $\{Y_{a_d}^{v_{id}}, Y_{a_d}^{v_{id}}, \ldots, Y_{a_d}^{v_{id}}\}$ of set $U$, where $Y_{a_d}^{v_{id}} = \{e \in U : f(e,a_d) = v_{id} \}$, $v_{id} \in V_{a_d}$ for $t=1,\ldots,d$, and $Y_{a_d}^{v_{id}} \cup \ldots \cup Y_{a_d}^{v_{id}} = U$, $Y_{a_d}^{v_{id}} \cap Y_{a_d}^{v_{id}} = \emptyset$ for $i \neq j$. The set $Y_{a_d}^{v_{id}}$ is called the $t$-th decision class (for $v_{id} \in V_{a_d}$).

Suppose that we have a set of positive examples:

$$S_{p}(Y_{v_i}^{a_i}) = \{e \in U : f(e,a_d) = v_{id} \}$$

and a set of negative examples:

$$S_{n}(Y_{v_i}^{a_i}) = \{e \in U : f(e,a_d) \neq v_{id} \text{ and } \forall e' \in S_{p}(Y_{v_i}^{a_i}) \exists a_j \in P, f(e,a_j) \neq f(e',a_j)\}$$

So, $S_{p}(Y_{v_i}^{a_i}) \cap S_{n}(Y_{v_i}^{a_i}) = \emptyset$ and $S_{p}(Y_{v_i}^{a_i}) \neq \emptyset$, $S_{n}(Y_{v_i}^{a_i}) \neq \emptyset$, by assumption; this assumption should hold in nontrivial problems, with larger data sets.

The rule IF $C^l$ THEN $[a_d = v_{id}^a]$ is called an "elementary" rule for class $Y_{v_i}^{a_i}$, where $C^l$ is a description of example in terms of attributes $a_j$, $j \in I$, and this example belongs to class $Y_{v_i}^{a_i}$. We consider the classification rules being the disjunction (via "\cup") of "elementary" rules consisting of complexes of type (2), i.e.

$$\text{IF } C^l \cup \ldots \cup C^l \text{ THEN } [a_d = v_{id}^a]$$

with: $I_1,\ldots,I_L \subseteq \{1,\ldots,K\}$, $C^l = \bigwedge_{j \in I} [a_j = v_j^{a_j}]$, $l = 1,\ldots,L$.

Suppose now that we have $P$ positive examples, $e^m \in S_{p}(Y_{v_i}^{a_i})$, $m = 1,\ldots,P$, and $N$ negative examples, $e^n \in S_{n}(Y_{v_i}^{a_i})$, $n = 1,\ldots,N$.

For each $a_j$, each possible value occurs at some intensity (frequency). If a value occurs more frequently in the positive examples and less frequently in the negative ones, then it should rather appear in the rule sought. These frequencies should clearly be relative.

This rationale may be formalized as: we introduce the function, for each $a_j$, $j = 1,\ldots,K$ and $v \in V_{a_j}$

$$g_j(v) = \frac{1}{P} \sum_{m=1}^{P} \delta(e^m,v) - \frac{1}{N} \sum_{n=1}^{N} \delta(e^n,v)$$

for each $v \in V_{a_j}$, where:

$$\delta(e^m,v) = \begin{cases} 1 & \text{for } v_j^{a_j} = v \\ 0 & \text{otherwise} \end{cases}$$

and:

$$e^m \in S_{p}, v_j^{a_j} = f(e^m,a_j) \in V_{a_j};$$

and analogously for $\delta(e^n,v)$.

So, we may expresses to what degree (from [-1, 1]) the particular values $v \in V_{a_j}$ of attribute $a_j$ occurs more often in the positive than negative examples. This idea is clearly more applicable to larger data sets.

We assume that $g_j(v) \in [-1,1]$ is used as a weight of value $v \in V_{a_j}$ of each $a_j$ (cf. Kacprzyk and Szkatula [19, 20]). An example $e_w$ with weights is written as

$$e_w = \bigwedge_{j=1}^{K} [a_j = v_j^{a_j}; g_j(v_j^{a_j})]$$

i.e. is a conjunction of weighted selectors, $s_j^w = [a_j = v_j^{a_j}; g_j(v_j^{a_j})]$, i.e.

$$C^l_w = \bigwedge_{j \in I_1 \cup \ldots \cup I_K} s_j^w$$

and is called a weighted complex.

Notice that for this $C^l_w$ vector $x$ has the elements $x_j = 1$ for $j \in I$, while, for $j \in \{1,2,\ldots,K\} \setminus I$, $x_j = 0$.

For a $C^l_w$ its weighted length is:
\[ d_w(C^I_W) = \sum_{j \in I} (1 - g_j(v^{a_j}_i)) \cdot x_j + =\]
\[ + (1 - g_j(v^{a_j}_i)) \cdot x_j = \sum_{j \in \{1,2,\ldots,K\}} \sum_{l=1}^K (1 - g_j(v^{a_j}_i)) \cdot x_j \]

which reflects a higher relevance of those values of attributes which occur more often in the positive than in negative examples.

The length of the weighted classification rule of \(L\) weighted complexes, \(R_w = C^I_W \cup \ldots \cup C^I_W\) is

\[ d_{R_0}(C^I_W \cup \ldots \cup C^I_W) = \max_{l=1,\ldots,L} d_w(C^I_W) \quad (10) \]

The problem of is to find an optimal classification rule \(R^*_w = C^I_W^* \cup \ldots \cup C^I_W^*\) such that

\[ \min_{l_1,\ldots,l_L} d_{R_0}(C^I_W^1 \cup \ldots \cup C^I_W^L) \quad (11) \]

As the (exact) solution of (11) is very difficult, an auxiliary problem is solved (cf. Kacprzyk and Szkatula [16]), i.e. an \(R_w = C^I_W^+ \cup \ldots \cup C^I_W^*\) is sought such that

\[ \min_{l_1} d_w(C^I_W), \ldots, \min_{l_L} d_w(C^I_W) \quad (12) \]

The solution of (12) is in general very close to that of (11), while much easier to obtain.

III. SOLUTION BY USING THE IP2-GA METHOD

We have a finite set of examples \(U\) [cf. (1)] and a finite set of attributes \(\{a_1,a_2,\ldots,a_K\} \cup \{a_d\}\), \(a_d\) is a decision attribute, \(V_{a_d} = \{v^{a_d}_1,\ldots,v^{a_d}_d\}\) is a domain of \(a_d\). We have the decision classes \(\{Y_v\}_{v \in V_{a_d}}: t = 1,\ldots,d\) for \(v \in V_{a_d}\). We have a set of positive examples \(S_p\), \(e^p \in S_p\), \(p = 1,\ldots,P\) and a set of negative ones \(S\\_n\), \(e^{p+n} \in S\\_n\), \(n = 1,\ldots,N\), \(S_p \neq \emptyset\), \(S_p \cap S\\_n = \emptyset\).

The problem: to find an optimal classification rule \(R^*_w = C^I_W^+ \cup \ldots \cup C^I_W^*\)

\[ I^*_1, I^*_2, \ldots, I^*_L \subseteq \{1,\ldots,K\}, \quad \text{minimizing the weighted length of the classification rule, can be represented as a modification of the set covering problems (Kacprzyk and Szkatula [16]).} \]

For \(e^p \in S_p\) and all the negative examples \(e^{p+n} \in S\\_n\), \(n = 1,\ldots,N\), we construct a 0-1 matrix \(Z_{N+K}[x_{nj}]\), \(n = 1,\ldots,N\), \(j = 1,\ldots,K\), defined as

\[ z_{nj} = \begin{cases} 
1 & \text{for } f(e^p, a_j) = f(e^{p+n}, a_j) \\
0 & \text{for } f(e^p, a_j) \neq f(e^{p+n}, a_j) 
\end{cases} \quad (13) \]

Its rows correspond to the consecutive negative examples \(e^{p+n} \in S\\_n\), \(n = 1,\ldots,N\) and its columns to the subsequent attributes \(a_1,\ldots,a_K\); \(z_{nj} = 1\) occurs if \(a_j\) takes on different values in the positive and negative example, i.e. \(f(e^p, a_j) \neq f(e^{p+n}, a_j)\), and \(z_{nj} = 0\) otherwise. There are clearly no rows with all elements equal 0 since the sets of positive and negative examples are disjoint (and non-empty). Thus, for any positive and negative example there always exists at least one attribute taking on a different value in these examples.

Consider now the following inequality

\[ \sum_{j=1}^K z_{nj} x_j \geq \gamma_n, \quad n = 1,\ldots,N \quad (14) \]

where \(\gamma = [\gamma_1,\ldots,\gamma_N]^T\) is a zero-one vector, and \(x = [x_1,\ldots,x_K]^T\) such that \(x \in \{0,1\}\) for \(j = 1,\ldots,K\).

Any vector \(x\) which satisfies \(Z x \geq \gamma\) (14) determines therefore in a unique way some complex composed of selectors from the description of the example such that the conditions of partial completeness and partial consistence are satisfied. It describes at least one example from the set of positive examples, and it does not describe most of the examples from the set of negative examples. If vector \(x\) does not describe the \(n\)-th negative example, then \(\gamma_n = 1\); and \(\gamma_n = 0\) otherwise.

The minimization in (12), using inequality (14), is

\[ \min_{x:Zx \geq \gamma} \sum_{j=1}^K (1 - g_j(v^{a_j}_i)) \cdot x_j \quad (15) \]

The minimization over the set of indices \(I^*_l\) may be replaced by the minimization with respect to \(x\) which yields [cf. (9)] an \(R^*_w = C^I_W^1 \cup \ldots \cup C^I_W^L\) such that

\[ \min_{x:Z^1x \geq \gamma} \sum_{j=1}^K (1 - g_j(v^{a_j}_i)) \cdot x_j \quad x:Z^Lx \geq \gamma \quad (16) \]

Each minimization with respect to \(x\) in (16) is therefore equivalent to the determination of a 0-1 vector \(x^*\) which uniquely determines the complex of the shortest weighted length. On the other hand, the satisfaction of \(Z^1x \geq A\) (\(A\) is a unit vector) guarantees that such a complex would not describe all negative examples. If rules defining class \(Y_{v^{a_d}_t}\) must describe almost none of
the negative examples, problem (15) can be written as a modification of the set covering problem

$$\min_{x, \gamma} \sum_{j=1}^{K} c_j x_j$$  \hspace{1cm} (17)  

$$\sum_{j=1}^{K} z_{nj} x_j \geq \gamma_n, \quad n = 1, \ldots, N$$  \hspace{1cm} (18)  

with an additional constraint

$$\sum_{n=1}^{N} \gamma_n \geq N - \text{rel}$$  \hspace{1cm} (19)  

where $$c_j = (1 - g_j(v_i^a j)), \quad z_{nj} \in \{0,1\}, \quad x_j \in \{0,1\}, \quad j = 1, \ldots, K, \quad \gamma = [\gamma_1, \ldots, \gamma_N]^T, \quad \gamma_n \in \{0,1\},$$ given a parameter $$\text{rel} \geq 0.$$  

This is the same as the original set covering problem with the exception that no more then $$\text{rel}$$ rows are uncovered. Then, clearly that no more then $$\text{rel}$$ rows can be deleted from the problem. We may, in deleting rows, lose some information about the problem that could have been better used. This reduction cannot always be applied. In the set covering problem (cf. Beasley and Chu [4]) there is only constraint (18), and $$\gamma = [\gamma_1, \ldots, \gamma_N]^T$$ is a unit vector.

The (17) – (19) is the problem of covering at least $$N \text{-rel}$$ rows of an $$N$$-row, $$K$$-column, zero-one matrix $$(z_{nj})$$ by a subset of the columns at minimal cost $$c_j.$$  

We define $$x_j = 1$$ if column $$j$$ with cost $$c_j > 0$$ is in the solution, and $$x_j = 0$$ otherwise. Equations (18) and (19) ensures that most rows (at least $$N \text{-rel}$$ rows) are covered by at least one column. It always has a feasible solution (a unit vector $$x$$ of $$K$$ element), due to the required disjointness of the sets of positive and negative examples and the way the matrix $$Z$$ was constructed.

So, we look for a 0-1 vector $$x$$ at minimum cost and a 0-1 vector $$\gamma = [\gamma_1, \ldots, \gamma_N]^T$$ which determines the covered rows, $$\gamma_n = 1$$ if $$n$$-th row is covered by solution $$x,$$ and $$\gamma_n = 0,$$ otherwise. By assumption, at least $$N \text{-rel}$$ rows must be covered by solution $$x.$$  

The set covering problem is a well-known combinatorial optimization problem and is NP-complete. A number of optimal and faster heuristic algorithms have been proposed, cf. Grossman and Wool [10]. Beasley and Chu [4] presented a genetic algorithm, with modified operations.

For the solution of (16) we propose a new procedure, IP2_GA, based on a genetic algorithm. We assume that the classification rules must correctly describe most of the examples, at least $$A_{\text{learning}};$$ the measure of classification accuracy $$A_{\text{learning}}$$ is the percentage of examples correctly classified.

We assume a $$K$$-bit binary string which represents a potential solution structure, where $$K$$ is the number of variables (i.e. columns in the set covering problem). The value 1 for the $$j$$-th bit implies that column $$j$$ is in solution $$x',$$ i.e. that $$x'_j$$ is in the solution.

In IP2_GA in each iteration all solutions are evaluated with respect to their completeness and consistency. We adopt a simple approach of using a penalty evaluation function which assigns utility to candidate solutions. The fitness of an individual solution $$x$$ is

$$\text{eval}(x) = f(x) - g \max \frac{K}{N} \sum_{n=1}^{N} f_n(x)$$  \hspace{1cm} (20)  

where: $$f(x) = \sum_{j=1}^{K} c_j x_j ; \quad f_n(x) = \begin{cases} 0 & \text{for } \sum_{j=1}^{K} z_{nj} \cdot x_j > 0 \\ 1 & \text{for } \sum_{j=1}^{K} z_{nj} \cdot x_j = 0 \end{cases}$$  

$$g = \max \{g_j; \quad j = 1, \ldots, K\}, \quad n = 1, \ldots, N,$$ where $$x_j$$ is the value of the $$j$$-th column in the string corresponding to the solution $$x$$ and $$c_j$$ is the cost of $$j$$-th column.

The structure of a new population is chosen by a stochastic universal sampling (cf. Baker [1]) with the wheel spun with a number of equally spaced markers equal to the population size.

The consecutive steps of IP2_GA are:

**Step 1.** Set the initial values: $$S = S_P,$$ i.e. the whole set of examples is initially assumed to contain the positive ones, $$S_N$$ is a set of negative examples, and $$R_{\text{init}}^* = \emptyset,$$ i.e. the initial set of complexes is assumed empty, iteration $$j = 0,$$ given parameter $$\text{rel} \geq 0.$$  

**Step 2.** Iteration $$j = j + 1.$$ Determine the weights $$G$$ by analyzing (preprocessing) of the examples due to (6).

**Step 3.** Determine an appropriate starting point; a good starting point may be a so-called centroid (cf. Kacpryzk and Szkatula [96]) that is some (possibly non existing) example in which the attributes take on values that occur most often in the positive examples and seldom in the negative examples. In the set of positive examples we find the closest positive example $$e'$$ to centroid, as the starting point for the next iterations.
Step 4. For the $e^p$ we form the matrix $Z_{N\times K}=[e_{nj}]$, $n=1,...,N$, $j=1,...,K$, due to (13) and form a modification of the set covering problem, due to (17), (18), (19).

Step 5. We apply a genetic algorithm.

Step 1'. Set $t=1$. Generate an initial population of random solutions $P(t)=\{x^1,x^2,...,x^P\}$. Each solution is simply a binary string of length $K$. Evaluate the fitness $eval(x^l)$ of individuals in the population, $l=1,2,...,P$.

Step 2'. For the first solutions a crossover operator is applied. Two solutions are chosen and form two new solutions.

Step 3'. A mutation operator is applied to each solution in the population.

Step 4'. The new solution generated by the crossover and mutation procedures may not be feasible. We evaluate the fitness $eval(x^l)$ of new individuals in the population.

Step 5'. If a termination condition is satisfied, then STOP, and the best solution is the one with the smallest fitness; otherwise, go to Step 6'.

Step 6'. Select a new population $P(t+1)$ from population $P(t)$ and return to Step 2'.

The 0-1 vector $x^*=[x^*_1,\ldots,x^*_K]^T$ found determines in a unique way the complex $C_{W}^{f_j}$ and the 0-1 vector $y=[y_1,\ldots,y_N]^T$ determines the fulfilled constraints. The complex can not describe more than $rel$ examples (given a parameter $rel \geq 0$). Now, we can go to Step 6.

Step 6. Include complex $C_{W}^{f_j}$ found in Step 5 into the classification rule sought $R_{W}^{*}$ (i.e. that with the minimal weighted length), $R_{W}^{*}:=R_{W}^{*} \cup C_{W}^{f_j} ; q(C_{W}^{f_j})$, where

$$q(C_{W}^{f_j}) = \frac{\text{number of examples covered by } C_{W}^{f_j}}{\text{number of all examples}}$$

and discard from the set of examples $S$ all examples covered by complex $C_{W}^{f_j}$.

Step 7. If the set of examples $S$ remaining is small enough, STOP and the rule

$$R_{W} = C_{W}^{f_{I_1}} ; q(C_{W}^{f_{I_1}}) \cup \ldots \cup C_{W}^{f_{I_L}} ; q(C_{W}^{f_{I_L}})$$

$I_1,\ldots,I_L \subseteq \{1,\ldots,K\}$, is the one sought; otherwise, return to Step 2.

The IP2_GA algorithm described above is relatively simple and efficient. It requires a number of parameters, e.g. the population size, probabilities of applying genetic operators, etc.

IV. APPLICATION OF THE IP2_GA ALGORITHM TO SOLVE A THYROID CANCER PROBLEM

For a lack of space, we only consider the example of a medical data set of Nakache and Asselian [23] that concerns 281 patients with thyroid cancer subjected to a surgery.

Each patient was described by 12 attributes: sex, {male, female}, age, {<40, 40-60, 60-70, >70} histology, {well differentiated, poorly differentiated}, metastasis, {yes, no}, enlargement, {uni-lobe, uni-lobe+isthm, all the thyroid}, clinical lymph nodes, {yes, no}, clinical aspect, {unique nodule, multi nodules, important enlargement}, pathological lymph nodes, {yes, no}, compressive syndromes, {yes, no}, invasion, {no, small, average, large}, survival time, {in months}, length in month of survival time from the entrance in the study (between 1960 and 1980) to the time of analysis, survival, {survivor, non survivor at time of analysis}. Two attributes are important: the survival time (in month) at the time of analysis, and the survival or non survival at the time of analysis.

The purpose is to find a prognostic rule for a new case coming from the same population and being in the same conditions.

We have two classes:

$\text{class 1: the patients will be alive over 7 years,}$

$\text{class 2: the patient will dead during 7 years.}$

and we seek two classification rules:

$\text{IF } R_{W}^{*} \text{ THEN [survival time over years] = 7}$

$\text{IF } R_{W}^{*} \text{ THEN [survival time below years] = 7}$

that must correctly describe most of the examples belonging to class 1 and 2, with at least $A_{learning}=97.5$.

The IP2_GA and IP2_GRE (with elements of a greedy algorithm) (Kacprzyk and Szkatuła [19, 20]) methods were applied.

The results of applying the methods to medical data are presented and described below.

TABLE 1. SOME PARAMETERS DESCRIBING THE PROCESS OF FINDING A CLASSIFICATION RULE FOR THE CLASS 1

\begin{tabular}{|c|c|c|}
\hline
Parameter & Value & Description \\
\hline
Population size & 100 & \\
\hline
Crossover probability & 0.8 & \\
\hline
Mutation probability & 0.05 & \\
\hline
\end{tabular}
TABLE 2. SOME PARAMETERS DESCRIBING THE PROCESS OF FINDING A CLASSIFICATION RULE FOR THE CLASS 2

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Number of iterations</th>
<th>Number of selectors in rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP2_GRE</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>IP2_GA</td>
<td>3</td>
<td>7</td>
</tr>
</tbody>
</table>

TABLE 3. SOME PARAMETERS DESCRIBING THE PROCESS OF CLASSIFICATION THE PATIENTS INTO THE FIRST OR THE SECOND CLASS

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>( A_{\text{learning}} ) %, by assumption</th>
<th>Classification accuracy, Achieved</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP2_GRE</td>
<td>at least 97.5 %</td>
<td>98.7 %</td>
</tr>
<tr>
<td>IP2_GA</td>
<td>at least 97.5 %</td>
<td>98.7 %</td>
</tr>
</tbody>
</table>

A classification rule for the first class is:

\[
\text{IF } [\text{metastasis} = \text{no}] \land [\text{clinical lymph nodes} = \text{no}] \land [\text{invasion} = \text{no}] \lor 0.468 \lor [\text{compressive syndromes} = \text{no}] \land [\text{invasion} = \text{average}] \lor 0.052 \lor [\text{clinical aspect} = \text{unique nodule}] \land [\text{clinical lymph nodes} = \text{yes}] \lor 0.104 \]

\[\text{THEN } [\text{survival time} = \text{over 7 years}]\]

and the shortest classification rules were obtained by using the method IP2_GA.

V. CONCLUDING REMARKS

We proposed an improved inductive learning procedure IP2_GA with elements of a genetic algorithm to derive classification rules sets of positive and negative examples. Results seem to be very encouraging.

Bibliography


