Experimental Study of Evolutionary Based Method of Rule Extraction from Neural Networks in Medical Data

Urszula Markowska-Kaczmar and Rafal Matkowski

Wroclaw University of Technology
Medical University of Wroclaw Poland
urszula.markowska-kaczmar@pwr.wroc.pl

Abstract. In the paper the method of rule extraction from neural networks based on evolutionary approach, called GEX, is presented. Its details are described but the main stress is focussed on the experimental studies, the aim of which was to examine its usefulness in knowledge discovery and rule extraction for classification task of medical data. The tests were made using the well-known benchmark data sets from UCI, as well as two other data sets collected by Lower Silesian Oncology Center.

1 Introduction

Neural networks (NN) are widely used in many real problems. They have become so popular because of their ability to learn from data instead to perform strictly the algorithm, which is sometimes difficult to define or to implement. During processing new data they can generalize knowledge they achieved in training procedure. Their ability to remove noise from data is well known, as well.

But there is a big disadvantage of neural networks (NN), which arrest the development of applications based on neural networks in many domains. It is the lack of ability to explain in what way they solve the problem. The medicine is an example of such a domain where the explanation of the final decision is very important in a computer supporting system based on neural network. The rise of the user trust is the main reason of development of the methods of knowledge extraction from neural networks. A brief survey of existing methods, their advantages and drawbacks are presented in the next section

The main part of the paper presents the method of rule extraction called GEX. The main emphasis is focused on the experimental study performed with the application of the method. They have two reasons. The first one was to test its skill to describe the performance of neural network solving the medical classification problem. The tests were made on the benchmark data sets from UCI and the results are compared to other methods.

GEX is developed in this way that by the setting its parameters it is possible to influence on the coverage of examples by a given rule. Rules that cover less examples but more than the value indicated by the user can contain new knowledge. An evaluation of the ability of GEX in this area was the second reason of

P. Perner (Ed.): ICDM 2006, LNAI 4065, pp. 76–90, 2006.

the experimental study. The evaluation of novelty needs the help of an expert so these tests were made on the data collected by Lower Silesian Oncology Center and in cooperation with its expert.

The paper is organized as follows. At the beginning the problem of rule extraction from neural network is presented. It creates the background for the description of GEX, which is presented in the next section. Then the experimental study is shown. Its first part is dedicated to the experiments testing the power of GEX in searching rules describing classification task made by a neural network. The second one investigates its ability in knowledge discovery.

2 The Problem of the Rule Extraction from Neural Networks

The typical feedforward neural network is presented in Fig. 1. Neurons in this network create layers. One neuron calculates the total activation (*net*) as the sum of the weighted signals that reach it and transforms it by the activation function f, which is usually nonlinear. In each layer information is processed in parallel, so it is difficult to describe in which way the network produces the final response. Knowledge about the problem which is solved by a neural network lies in its architecture, and the parameters: weights assigned to the connections, activation functions, biases and in the set of training patterns. That is why all these elements are considered in the rule extraction methods.

The taxonomy distinguishes two main approaches. The global methods treat a neural network as a black box and in the searching rules they use the patterns processed by the network. We can mention here: KT [1], NeuroRule [2], Partial and Full-Re [3] or for regression problem - [4].

The second group describes the activity of each neuron in the form of a rule and by aggregation of these rules the set of rules specifying the performance of the trained neural network is obtained. Between these methods we can cite methods from: [5], [3], [6]. From this short survey one can notice that many methods of rule extraction exist. They differ from each other on the achieved

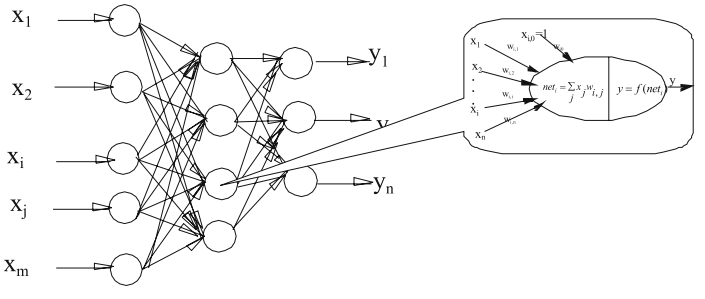

Fig. 1. The scheme of a feedforward neural network with detailed operations of one neuron

results. Some of them are dedicated to the special type of the neural network, some need a retraining of the neural network during the rule extraction or a special rule of the neural network training or they are dedicated to the special type of neural network attributes, so the need to design the method that are free from the above mentioned disadvantage still exists.

Andrews [7] has formulated the following criteria that allow to evaluate acquired set of rules.

- *fidelity* – expresses the way, in which the set of rules mimics the neural network performance;
- *accuracy*– describes the quality of new patterns classification;
- *consistency* – it exists when during different rule extraction session the produced sets of rules give the same classification;
- *comprehensibility* – is expressed in terms of the number of rules and the number of premises in the rules.

In real applications the weight of each criterion can be different. Citing after [6] suitable algorithm of the rule extraction should posses the following features: it should be independent of the architecture of neural network, it should not require its retraining and it should characterise by high accuracy and fidelity.

In the paper the problem of knowledge extraction from a neural network is formulated as follows. The trained neural network that solves classification task and the set of training patterns are given. The designed method should find a set of prepositional rules, that describes the performance of this neural network satisfying the criteria given by Andrews. Other representation of the neural network description are also used, for example decision trees [8], but because of the comprehensibility we focus on the prepositional rules that take the following form:

$$IF\ premise_1\ AND\ premise_2 ... premise_n\ THEN\ class_v, \tag{1}$$

the $i-$th premise corresponds to the $i-$th neural network input. The premise specifies a condition put on the values of the input attribute of neural network to satisfy the rule. After THEN stands a conclusion, which is unambiguously defined by the label of the class. The relationship between neural network and the rule is shown in Fig. 2.

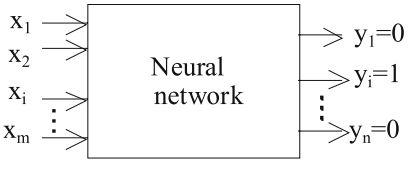

Fig. 2. The relationship between the rule and the neural network

In the classification problem the output of neural network is locally encoded. It means that to designate i-th class only i-th output is equal to 1, the remaining outputs are equal to 0.

Taking into account the number of the neural network inputs and the type of attributes that can be not only binary but nominal or real one, searching for some limitations in premises of the rules can be seen as the NP - hard problem. That is why evolutionary approach can be useful in this case. The idea is not new [9]. Unfortunately, the level of complexity of this problem prevents the application of a simple genetic algorithm, so existing methods applying a genetic algorithm differ in the way of coding and obtaining the final set of rules [10], [11], [12].

3 The Basic Concepts of GEX

In GEX the formation of species by simultaneously evolving subpopulations is introduced (Fig. 3). The individuals in subpopulation can evolve independently or optionally migration of individuals is possible. Each species contains individuals corresponding to one class, which is recognized by the NN. One individual in a subpopulation encodes one rule. The form of the rule is described by (1). The premise in a rule expresses a condition, which has to be satisfied by the value of the corresponding input of the neural network in order to classify the pattern to the class indicated by the conclusion of the rule. The form of the premise is depending on the type of attribute, which is included in the pattern. In practice the i-th pattern is identified by the vector x_i (2):

$$\mathbf{x}_i = [x_{i,1}, x_{i,2}, ..., x_{i,n}], \tag{2}$$

where $x_{i,j}$ is the value of the attribute (feature) X_j. Each pattern is the element of Cartesian product:

$$d(X_1) \times d(X_2) \times ... \times d(X_n) \tag{3}$$

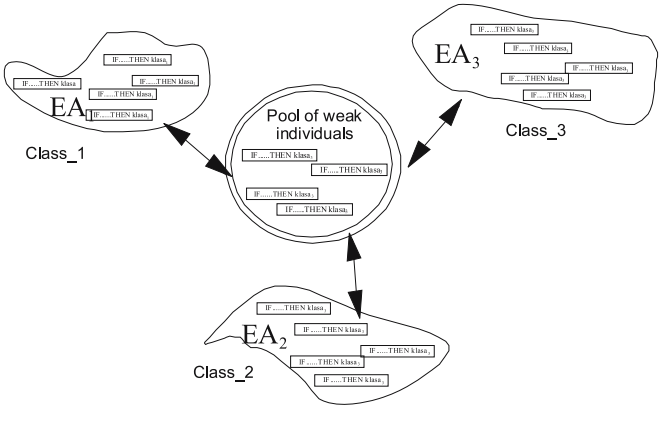

Fig. 3. The idea of GEX method

and $d(X_j)$ is the domain of the feature X_j.
In GEX we concern the following types of attributes:

- *real* $X_j \in V_r \Rightarrow X_j \in \Re$.
  Between them two types are distinguished:
  - *continuous* - V_c: their domain is defined by a range of real numbers: $X_j \in V_c \Leftrightarrow d(X_j) = (x_{jmin}; x_{jmax}) \in \Re$.
  - *discrete* V_d: the domain creates a countable set W_d of values w_i and the order relation is defined on this set
    $X_j \in V_d \Leftrightarrow d(X_j) = \{w_i \in \Re, i = 1, ...k, k \in \aleph\}$.
- *nominative* V_w: the domain is created by a set of discrete unordered values $X_j \in V_w \Leftrightarrow d(X_j) = \{w_1, w_2, ...w_w\}$, where w_i is a symbolic value.
- *binary*V_b: the domain is composed of only two values *True* and *False* $X_j \in V_b \Leftrightarrow d(X_j) = \{True, False\}$.

A condition in the premise differs depending on the type of the attribute. For a real type of the attribute (discrete and continuous) the following premises are covered:

- $\Rightarrow x_i < value_1$,
- $\Rightarrow x_i < value_2$,
- $\Rightarrow x_i > value_1$,
- $\Rightarrow x_i > value_2$,
- $\Rightarrow value_1 < x_i \bigwedge x_i < value_2$,
- $\Rightarrow x_i < value_1 \bigvee value_2 < x_i$.

For a discrete attribute, instead of $(<, >)$ inequalities $(\leq, \geq)$ are used.
For enumerative attributes – only two operators of relation are used $\{=, \neq\}$, so the premise has one of the following form:

- $x_i = value_i$,
- $x_i \neq value_i$.

For boolean attributes there is only one operator of relation $=$. It means that the premise can take the following form:

- $x_i = True$,
- $x_i = False$.

All rules in one subpopulation have identical conclusion. The evolutionary algorithm (EA) is performed in a classical way (Fig. 4).

First, the initial population is created. Then, the individuals are evaluated and the best rules are the candidates to send to the final set of rules that describes the performance of the neural network. They become a members of this set when they are more general than the rules existing in this set. It means that the less general rules are removed from it. Next, by the selection of individuals from the current population and after applying genetic operations (crossover, mutation and optionally migration) the offspring population is created.

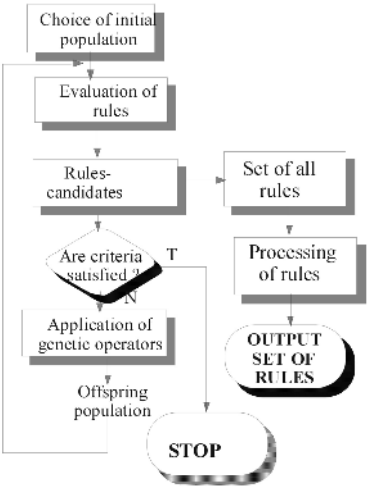

Fig. 4. The schema of evolutionary algorithm in GEX

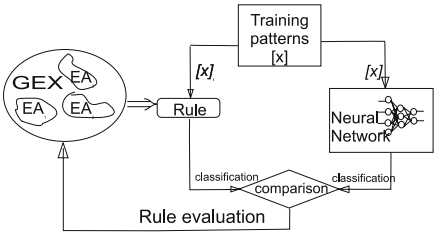

Fig. 5. The rule evaluation in GEX

It can be noticed that the only difference between classical performance of evolutionary algorithm and the proposed one lies in the evaluation of individuals, which requires the existence of decision system based on the rule processing. In each generation (after decoding) rules are evaluated by the comparison of the neural network answer and classification of patterns made upon the rules (Fig. 5). To realize it a decision system consisting in searching the rule that covers the given pattern is implemented. Classification made by the neural network serves as an oracle for the evaluated rules. The comparison of the results of classification is the basis for the evaluation of each rule, which is expressed by the value of a fitness function. Evolutionary algorithm performing in the presented way will look for the best rules that cover as many patterns as possible. In this case the risk exists that some patterns never would be covered by any rule. To solve this problem in GEX the niche mechanism is implemented. The final set of rules is created on the basis of the best rules found by evolutionary algorithm but also some heuristics are developed in order to optimize it.

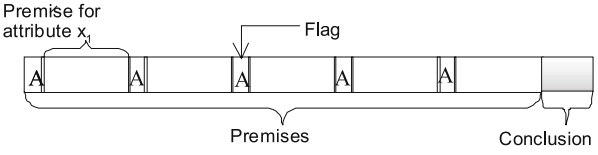

Fig. 6. Scheme of a chromosome in GEX

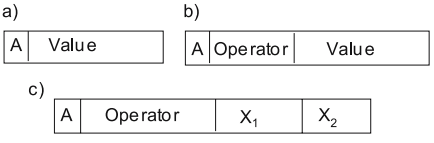

Fig. 7. The designed genes in GEX

3.1 Evolutionary Algorithm

To apply evolutionary algorithm the following elements are essential to design: representation of solution in the genotype, genetic operators and a fitness function.

The Genotype. Figure 6 shows the general scheme of the genotype in GEX. It is composed of the chromosomes corresponding to the inputs of neural network and a single gene of conclusion. A chromosome consists of gene being a flag and genes encoding premises, which are specific for the type of attribute of the premise it refers to.

The existence of flag assures that the rules have a different length, because the premise is included in the body of the rule if the flag is set to 1, only. In order to reflect the condition in the premise the chromosome is designed dependently on the type of attribute (Fig.7). For the real type of attribute the chromosome consists of the code of relation operator and two values determining the limits of range (Fig.7c). For the nominal attribute there is a code of operator and value (Fig.7b). Figure 7a represents a chromosome for the binary attribute. Besides the gene of flag, it consists of one gene referring to the value of attribute.

Selection and Genetic Operators. The initial population is created randomly with the number of individuals equal to $StartSize$. The basic operators used in GEX are a crossover and a mutation. They are applied after a selection of individuals that creates a pool of parents for the offspring population. In the selection a roulette wheel is used. The individuals that are not chosen to become parents are moved to the pool of weak individuals (Fig. 3). In each generation the size of a population is decreased by 1. When the population size reaches the value defined by the parameter $MinSize$ migration operator becomes active. It consists in taking individuals from the pool of weak individuals (Fig. 3) to increase the size of the population to $Nsize$. In case the migration is inactive a kind of macromutation is used.

Although in the application of GEX we can choose between one point, two points and uniform crossover in the presented experiments the two-points crossover was used. It relies on the choice of a couple of the parent genotypes with the probability p_{c-w}, then two points are chosen in random and information is exchanged. These points can only lie between chromosomes. It is not allowed to cut the individuals between genes in the middle of the chromosome.

The mutation is specifically design for each type of a gene and is strongly dependent on the type of the chromosome (premise) it refers to. It changes information contained in the gene. The following parameters define this operator:

- p_{mu-op} - the probability of mutation of the relation operator or binary value,
- $p_{mu-range}$ - the probability of mutation of the range limits,
  p_{mu-act} - the probability of mutation of value for genes in chromosomes for nominative attributes,
- r_{ch} - the change of the range.

The mutation of the flag A relies in the change of its actual value to the opposite one with probability p_{mu-op}. The mutation of the gene containing *value* in the chromosome of the binary attribute is realized as the change of the gene *value* to its opposite value with the probability p_{mu-op} (True to False or False to True). The mutation of the gene *Operator* independently of the chromosome consists in the change of the operator to other operator defined for this type of premise with the probability p_{mu-op}. The mutation of gene referring to the *value* in the chromosomes for the nominative attribute is realized as the change of the actual value to the other one specified for this type with the probability p_{mu-act}. The mutation of the gene encoding the limits of a range in chromosomes for the real attributes consists in the change of $value_1$ and $value_2$. It is realized distinctly for continuous and discrete values. For continuous attributes the limits are changed into new values by adding a value from the following range (4).

$$(-(x_{imax} - x_{imin}) \cdot r_{ch}; (x_{imax} - x_{imin}) \cdot r_{ch}), \tag{4}$$

where x_{imax} and x_{imin} are respectively the maximal and minimal values of i-th attribute, r_{ch} is the parameter, which defines how much the limits of range can be changed. For the discrete type the new value is chosen in random from the values defined for this type.

Fitness Function. The assumed fitness function, is defined as the weighted average of the following parameters: accuracy (*acc*), classCovering (*classCov*), inaccuracy (*inacc*), and comprehensibility (*compr*):

$$Fun = \frac{A * acc + B * inacc + C * classCov + D * compr}{A + B + C + D} \tag{5}$$

Weights (A, B, C, D) are implemented as the parameters of the application. *Accuracy* measures how good the rule mimics knowledge contained in the neural network. It is defined by (6).

$$acc = \frac{correctFires}{totalFiresCount}, \tag{6}$$

where *totalFiresCount* is the number of patterns covered by the evaluated rule, *correctFires* is the number of patterns covered by the rule that are classified by the neural network in the same way as specifies the conclusion of evaluated rule. *Inaccuracy* is a measure of incorrect classification made by the rule. It is expressed by eq. (7).

$$inacc = \frac{missingFires}{totalFiresCount} \tag{7}$$

Parameter *classCovering* contains information about the part of all patterns from a given class, which are covered by the evaluated rule. It is formally defined by eq. (8);

$$classcov = \frac{correctFires}{classExampelsCount}, \tag{8}$$

where *classExamplesCount* is a number of patterns from a given class.
The last parameter - *comprehensibility* is calculated on the basis of eq. (9).

$$compr = \frac{maxConditionCount - ruleLength}{maxConditionCount - 1}, \tag{9}$$

where *ruleLength* is the number of premises of the rule, *maxConditionsCount* is the maximal number of premises in the rule. In other words, it is the number of inputs of the neural network.

3.2 The Set of Rules

During an evolution the set of rules is updated. Some rules are added and some are removed. In each generation individuals with *accuracy* and *classCovering* greater than *minAccuracy* and *minClassCovering* are the candidates to update the set of rules. The values *minAccuracy* and *minClassCovering* are the parameters of the method.

The rules are added to the set of rules when they are more general than the rules actually being in the set of rules. Rule r_1 is more general than rule r_2 when the set of examples covered by r_2 is a subset of the set of examples covered by r_1. In case the rules r_1 and r_2 cover the same examples, the rule that has the bigger fitness value is assumed as more general one. Furthermore, the less general rules are removed. After presentation of all patterns for each rule *usability* is calculated according to eq.(10).

$$usability = \frac{usabilityCount}{examplesCount} \tag{10}$$

All rules with *usability* less then *minUsability*, which is a parameter set by the user, are removed from the set of rules. We can say that optimization of the set of rules consists in removing less general and rarely used rules and in the supplying it by more general rules from the current generation.

The following statistics characterize the quality of the set of rules. The value *covering* defines the percentage of the classified examples from all examples used in the evaluation of the set of rules (eq. 11).

$$covering = \frac{classifiedCount}{examplesCount} \tag{11}$$

Fidelity expressed in (eq.12) describes the percentage of correct (according to the neural network answer) classified examples from all examples classified by the set of rules.

$$fidelity = \frac{correctClassifiedCount}{classifiedCount} \tag{12}$$

Covering and $fidelity$ are two measures of quality of the acquired set of rules that say about its accuracy generalization. Additionally, the $performance$ (eq.13) is defined, which informs about the percentage of the correct classified examples compared to all examples used in the evaluation process.

$$performance = \frac{correctClassifiedCount}{examplesCount} \tag{13}$$

4 Experimental Studies

The experimental studies have two aims. First, its efficiency in describing classification decision made by the neural network on the medical data was tested. In these experiments we used the data sets collected in UCI repository [13]. The results are compared with other known methods of the rule extraction.

The second series of experiments was made with using the data sets collected by Lower Silesian Oncology Center. The first one contains 527 records of patients with *Primary cancer of the cervix uteri*, the second one contains 101 records describing patients with *Ductal breast cancer* treated in this Oncology Center. They are described in subsection 4.2.
On the basis of the preliminary experiments with GEX we observed that one can influence on the set of acquired rules by:

- the fitness function (the part *comprehensibility* - as shorter the rule is - the more general it is, the shorter set of rules we obtain in the consequence),
- the assumed value of *minaccuracy*, (a value less than 1, allows to acquire rules that cover more patterns but some of them are covered incorrectly),
- the value of *minusability* parameter – its value defines the minimal number of the covered patterns by each rule to become a member of the final set of rules. When it is high we expect to obtain very general rules.

In classification task we are interested in acquiring rules that are very general. It means, they cover many patterns with high accuracy. It is in contrast to knowledge discovery, when we are looking for rules that cover less patterns but the rules point at new knowledge, so novelty is essential in this case. The second goal of our experiments was to test possibility in application of GEX to knowledge discovery. Because the novelty of acquired knowledge has to be evaluated we use the data from Oncology Center and a help of an expert.

Table 1. The result of experiments of GEX with assumed performace=98% using $10-cross\ validation$; NG - number of generations, NR - number of rules for files from UCI repository with different types of attributes

file	*NR*	*NG*	*covering*	*fidelity*
Breast Cancer	18,6± 2,04	61,8± 29,9	0,975±0,022	0,982±0,018
WDBC	27,52± 4,19	1789,8± 191,4	0,486±0,125	0,968±0,031
Pima	28.36± 3.04	1477± 332.4	0,81 ±0,099	0,975±0,022
Liver	31,92± 4,01	1870,9± 121,5	0,173±0,102	0,674±0,291
Dermatology	20,24± 2,76	949,3± 452,3	0,829± 0,067	0,981±0,022
Heart	28.36± 3.04	1477± 332.4	0,921±0,048	0,836±0,089
Hypothyroid	21.96± 20.67	316.0± 518.9	0.960±0.048	0.996±0.004

Table 2. The comparison of GEX and *NeuroRule* on the *Breast cancer* data set

NeuroRule	GEX			
	Minusab=1		Minusab=10	
Accuracy	Accuracy	number of rules	Accuracy	number of rules
98,10	98,71± 0,0057	10.30± 2,31	97,5± 0,072	4,2± 0,63

4.1 The Ability of GEX to Describe Classification Made by the Neural Network

In the first experiment we applied GEX for the well known medical data from UCI [13] such as: *Breast Cancer*, *Wisconsin Breast Cancer*, *Liver*, *Hypothyroid*, *Heart*, *Dermatology*. The parameters of the method were as follows: p_{mu-op}=0.2, $p_{mu-range}$=0.2, p_{mu-act}=0.2, r_{ch}=0.1, niching=on, migration=on, weights in the fitness function: A=2, B=2, C=-2, D=1, p_{c-w}=0,5, $Nsize = startsize$=40, $minsize$ = 30 individuals, $minaccuraccy$=1 and $minusability$=1.

In the experiments the evolution was stopped when the set of acquired rules has reached the performance 98% or when during 250 generations there was no progress in the evolution. 10 *fold cross validation* was applied to evaluate the final set of rules. The results are shown in Table 1. For each file the first column in this table describes the number of the acquired rules (NR) in the final set of rules, the second one is the number of generations (NG) needed to reach this performance. The third and fourth columns refer to covering and fidelity, respectively. One can notice that independently of the type of attributes, that are contained in the experimental data GEX was able to extract rules. Let us emphasize, the aim of this experiment was not to search for the set with the minimal number of rules.

In order to compare the result of GEX to other methods, the experiments were repeated trying to keep the same conditions. Table 2 presents the comparison to the result of *NeuroRule* described in [2]. The *Breast Cancer* data set was split into two equal parts - the training and the testing sets. The quality of the set of rules was measured by its accuracy. Table 2 shows two results of GEX obtained with different parameters settings. With $MinUsabilty$=10 the average

Table 3. The comparison of GEX and *FullRe* on the *Breast cancer* data set

FullRe		GEX	
fidelity		fidelity	
training set	testing set	training set	testing set
96,77	95,61	98,36± 0,99	95,60± 0,87

number of acquired rules was equal to 4.2 and the accuracy was slightly smaller than for $Minusability$=1. Comparing both results to $NeuroRule$ one can say that accuracy is comparable. The number of rules for $NeuroRule$ was equal to 5 but this method assumes the default rule, which is used in case when none of the extracted rules could be fired.

The comparison with the results of FullRe [3] made on the *Breast cancer* data set is showed in Table 3. The data set is split fifty-fifty in the training and the testing set. The results for GEX are the average from 50 runs after 2000 generations. They were obtained with the parameters described above. The only difference was the value of weight D=10. The FullRe method, like NeuroRule, extracts rules using a default class rule. Taking into account the quality of acquired rules expressed by performance we can say that the results are comparable, but GEX deliver the description for each class.

4.2 The Ability of GEX to Acquire New Knowledge

The experiments described in this section were made on the basis of two data files from Lower Silesian Oncology Center. The first one comes from 5-year observation of 527 patients with primary cancer of the Cervix uteri treated in 1996, 1997 and 1998. The clinical and pathological data available on these patients include: the date of birth and the patients age, FIGO stage of the disease (according to FIGO Staging, 1994), tumor size, histological type of the tumor, the degree of differentiation of the tumor, interval between diagnosis and first treatment (both dates), the type of a surgical treatment, the type of a performed radiotherapy,the duration of radiotherapy, the assessment of the response to a treatment, the date of the end of hospitalization, the last known vital status or the date of death, the relapse-free survival, the overall survival.

The second data set contains 5-year observation of 101 patients with *Primary ductal breast cancer* (stage II) treated in 1993 and 1994. ER and nm23 expression was analyzed by immunohistochemical procedures. The other clinical and pathological data available on these patients included: Bloom and Richardson's grade, the tumor size, the status of axillary lymph nodes, the relapse-free survival, the overall survival, the body mass index, the hormonal status and several other data from anamnesis and family history.

The role of the specified parameters for both distinguished cases (classification and knowledge discovery) was examined on the basis of data with *Primary cancer of the cervix uteri*. In both data sets two classes were distinguished: the first one refers to the patients who after 5 years starting from the treatment were

Table 4. The result of experiments for different values of parameters

Parameters	Experiment1	Experiment2	Experiment3	Experiment4
D	20	10	8	6
Minaccuracy	0.8	1.0	0.95	0.95
Minusability	20	10	10	1
Number of rules	3	10	10	47
Total covering[%]	96	75	87	96.5
number of patterns in class$_1$				
correct covered	154	123	139	164
incorrect covered	36	0	17	11
uncovered	6	73	40	40
number of patterns in class$_2$				
correct covered	286	254	277	294
incorrect covered	22	0	3	11
uncovered	13	67	41	16

alive (for *Cervix uteri* data set 321 patterns), and the second class containing the patients who died ahead 5 years (for *Cervix uteri* data set 196). Table 4 presents the example of the results for different values of parameters. We can observe that the less is the value of *minusability*, the more rules arrives in the final set of rules. This phenomena is also connected with the weight D in the fitness function and *minaccuracy* (47 rules for *minusabilty*=1 and D=6 *minaccuracy*=0,95 but only 3 rules for *minusabilty*=20 and D=20, *minaccuracy*=0,8). The shorter is the rule, the more general it is, in consequence the less number of rules is needed to cover the patterns.

This statement gives the start point to the next step of experiment, where we tried to evaluate the extracted rules in the sense of knowledge they bring for the end user. To realize it we collected rules extracted in the experiment1 and experiment4 from the table 4 and gave to the expert for evaluation. In the same way we have extracted rules for the second data - *Ductal breast cancer*, as well.

For experiment1 the example of the rule for *Cervix uteri* data set is shown bellow:

IF $(DegreeOfDifferentiation >= 1,00$ *and* $DegreeOfDifferentiation <= 3,00)$ **AND** $histotype <> 1$ **AND** $(TimeDiagnosis-Treatment >= 50,28$ *and* $TimeDiagnosis - Treatment <= 586,28)$ **AND** $SurgicalCode = 3$ **AND** $(ResponseToTreatment >= 1,00$ *and* $ResponseToTreatment <= 2,00)$ **THEN** 1

The comment of the expert was as follows: this rule describes in the accurate way the factors of a good prognosis for a patient: the low illness advance, the radical surgical treatment and the effective radiotherapy gives a good chance to survive.

For all rules obtained in the experiment1 the comments of the expert was similar to the one above. It means that rules describe dependence between class and attributes in the way that is confirmed by the experience of the physician. It is very important aspect of GEX application because it can increase the trust

of the user to the system. From the other hand, these rules are not revealing. They contain general knowledge only.

To discover new knowledge the parameters of GEX described for experiment4 demonstrates their superiority. For example for the *Primary ductal breast cancer* data set the following rule was found:

IF$(age >= 20,00$ *and* $age <= 50,96)$ **AND** $(ER >= 3,00$ *and* $ER <= 11,00)$ **AND** $(sizeoftumor >= 0,00$ *and* $sizeoftumor <= 3,00)$ **AND** $(birthrate >= 1,00$ *and* $birthrate <= 2,00)$ **AND** $(numberoftreatments >= 0,00$ *and* $numberoftreatments <= 5,00)$ **AND** $(timeoftreatments >= 237,58$ *and* $timeoftreatments <= 3400,00)$ **THEN** 0

The rule was commented by the expert as follows: It is surprising. I would rather think, that the prognosis would be high because ER is positive. Since it refers to the relatively large number of patients it should be widely examined.

The experimental study confirms that by appropriate setting parameters of GEX method we can extract rules for the classification task made by the neural network but also GEX can be seen as a tool for knowledge discovery. For the first case of application of GEX we can suggest that the value of *minusability* and weight D should be high. In this case we obtain rules which are as general as possible, but it is rather difficult to expect they deliver nontrivial dependencies. For the less value of parameter D in the fitness function the rules with more number of premises arrive. This fact combined with the low value of *minusability* explains the high number of rules for the experiment4. We can filter rules from the acquired set rules that have sufficient support to give physicians.

5 Conclusion

The experiments have shown that by affecting on the parameters of the proposed method that control the number of examples covered by the rules GEX can be the useful tool to deliver rules describing classification task made by the neural network and also to discover dependence hidden in data processed by the neural network. In the paper GEX is compared to other methods. Its results are similar or even better comparing to other methods, but it has not default rule, giving in the consequence the description of classification for each class. In comparison to other rule extraction methods the novelty of GEX lies in the design of genotype that enables to process various types of attribute and heuristics that optimize the final set of rules.

Although the tests examining the consistency of GEX remain for the future, on the basis of the experiments that has been doing so far we can conclude that independence of the type of attributes, ability to control the number of the patterns covered by the rules in the final set of rules (that enables to use GEX in rule extraction for classification made by a neural network as well as for knowledge discovery), independence of the neural network architecture and nonexistence of default rule make GEX very attractive alternative to other rule extraction methods.

References

1. Fu, L.M.: Rule generation from neural network. IEEE. Transactions on Systems, Man and Cybernetics **vol. 24** (1994) 1114–1124
2. Lu, H., Setiono, R., Liu, H.: Neurorule: a connectionist approach to datamining. In: Proc. 21 st Conference on very Large Databases, Zurich. (1995)
3. Taha, I., Ghosh, J.: Symbolic interpretation of artificial neural networks. Technical report, The Computer and Vision Research Center, University of Texas, Austin (1996)
4. Setiono, R., Thong, J.: An approach to generate rules from neural networks for regression problems. European Journal of Operational Research **155** (2004) 239–250
5. Palade, V., Neagu, D.C., Patton, R.J.: Interpretation of trained neural networks by rule extraction. Fuzzy Days 2001, LNC 2206 (2001) 152–161 Springer-Verlag Berlin Heilderberg 2001.
6. Thrun, S.B.: Extracting rules from artificial neural networks with distributed representation, advances. Neural Information Processing Systems **vol. 7** (1995)
7. Andrews, R., Diederich, J., Tickle, A.: A survey and critique of techniques for extracting rules from trained neural networks. Knowledge-Based Systems **8** (1995) 373–389
8. Craven, M., Shavlik, J.: Extracting tree-structured representations of trained networks. Advances Information Processes Systems **Vol. 8** (1996.) MIT Press, Cambridge, MA.
9. Vinterbo, S. Ohno-Machado, L.: A genetic algorithm approach to multi-disorder diagnosis. Artificial Intelligence in Medicine **18** (2000) 117–132
10. Francisci, D., Brisson, L., Collard, M.: A scalar evolutionary approach to rule extraction. Technical report, ISRN I3S/RR-200312-FR (2003)
11. Fidelis, M., Lopes, H.S., Freitas, A.: Discovering comprehensible classification rules with genetic algorithm. In: Proc. Congress on Evolutionary Computation (CEC-2000). (2001) 805–810
12. Arbatli, D.A., Akin, L.H.: Rule extraction from trained neural network using genetic algorithm. Nonlinear Analysis, Theory Methods and Application **30** (1997) 1639–1648
13. Murphy, P.M., Aha, D.W.: UCI repository of machine learning databases. PhD thesis, Department of Information and Computer Science, University of California, Irvine, CA (1998)