Proceedings of the 2007 IEEE Symposium on
Foundations of Computational Intelligence (FOCI 2007)

Rough-Neuro-Fuzzy Systems for Classification

Krzysztof Cpałka, Robert Nowicki, Leszek Rutkowski, *Fellow*, IEEE♣

Abstract: In the paper we present flexible neuro-fuzzy systems and a method for their reduction. The method is based on the concept of the weighted triangular norms. Moreover, a rough-neuro-fuzzy classifier working in the case of missing features is described.

I. Introduction

In the literature various classification methods have been proposed (see e.g. [5]). Some of them are based on neural networks, fuzzy systems and rough sets (see e.g. [6]-[12]). It is well known that traditional fuzzy systems suffer from the lack of learning properties. On the other hand neural networks are not able to incorporate a linguistic information coming from human experts. Neuro fuzzy systems presented by several authors (see e.g. [4], [6]-[10],[13]-[18]) exhibit advantages of neural networks and fuzzy systems. In this paper we develop a new class of neuro-fuzzy systems. It is well known that introducing additional parameters to be tuned in neuro fuzzy systems improves their performance and they are able to better represent the patterns encoding in data. Therefore, in this paper we introduce several flexibility concepts in the design of neuro fuzzy systems. Due to additional parameters incorporated into a neuro fuzzy system, we achieve an excellent performance of the classification. Moreover, a procedure for reduction of flexible neuro-fuzzy systems will be presented and tested. A high accuracy of a neuro fuzzy classifier is demonstrated in simulation examples. Another classifier will be studied in the case of missing data. The rough-fuzzy sets are incorporated into Mamdani type neuro-fuzzy structures and the rough-neuro-fuzzy classifier is derived. An experiment illustrating the performance of the rough-neuro-fuzzy classifier working in the case of missing features will be described.

II. Flexible neuro-fuzzy systems

We consider multi-input, single-output neuro-fuzzy system mapping $\mathbf{X} \to \mathbf{Y}$, where $\mathbf{X} \subset \mathbf{R}^n$ and $\mathbf{Y} \subset \mathbf{R}$. The fuzzifier performs a mapping from the observed crisp input space $\mathbf{X} \subset \mathbf{R}^n$ to the fuzzy sets defined in $\mathbf{X}$. The most commonly used fuzzifier is the singleton fuzzifier which maps $\bar{\mathbf{x}} = [\bar{x}_1, \ldots, \bar{x}_n] \in \mathbf{X}$ into a fuzzy set $A' \subseteq \mathbf{X}$ characterized by the membership function

$$\mu_{A'}(\mathbf{x}) = \begin{cases} 1 & \text{if} \quad \mathbf{x} = \bar{\mathbf{x}} \\ 0 & \text{if} \quad \mathbf{x} \neq \bar{\mathbf{x}} \end{cases} \tag{1}$$

The fuzzy rule base consists of a collection of N fuzzy IF-THEN rules in the form

$$R^{(k)} : \text{IF } \mathbf{x} \text{ is } A^k \text{ THEN } y \text{ is } B^k \tag{2}$$

where $\mathbf{x} = [x_1, \ldots, x_n] \in \mathbf{X}$, $y \in \mathbf{Y}$, $A_1^k, A_2^k, \ldots, A_n^k$ are fuzzy sets characterized by membership functions $\mu_{A_i^k}(x_i)$, whereas B^k are fuzzy sets characterized by membership functions $\mu_{B^k}(y)$, respectively, $k = 1, \ldots, N$.

The fuzzy inference determines a mapping from the fuzzy sets in the input space $\mathbf{X}$ to the fuzzy sets in the output space $\mathbf{Y}$. Each of N rules (2) determines a fuzzy set $\bar{B}^k \subset \mathbf{Y}$ given by the compositional rule of inference

$$\bar{B}^k = A' \circ \left(A^k \to B^k\right) \tag{3}$$

where $A^k = A_1^k \times A_2^k \times \ldots \times A_n^k$, and

$$\begin{aligned} \mu_{\bar{B}^k}(y) &= \mu_{A_1^k \times .. \times A_n^k \to B^k}(\bar{\mathbf{x}}, y) \\ &= \mu_{\mathbf{A}^k \to B^k}(\bar{\mathbf{x}}, y) \\ &= I\left(\mu_{\mathbf{A}^k}(\bar{\mathbf{x}}), \mu_{B^k}(y)\right) \end{aligned} \tag{4}$$

where $I(\cdot)$ is an "engineering implication" (Mamdani approach) [7] or fuzzy implication [3].

Neuro-fuzzy architectures developed so far in the literature are based on the discretization of formula

$$\bar{y} = \frac{\int_{\mathbf{Y}} y \mu_{B'}(y) dy}{\int_{\mathbf{Y}} \mu_{B'}(y) dy} \tag{5}$$

by

$$\bar{y} = \frac{\sum_{r=1}^{N} \bar{y}^r \cdot \mu_{B'}(\bar{y}^r)}{\sum_{r=1}^{N} \mu_{B'}(\bar{y}^r)} \tag{6}$$

where $\bar{y}^r$ denotes centres of the membership functions $\mu_{B^r}(y)$, i.e. for $r = 1, \ldots, N$

$$\mu_{B^r}(\bar{y}^r) = \max_{y \in \mathbf{Y}} \{\mu_{B^r}(y)\} \tag{7}$$

It has been always assumed that number of terms in formula (6) is equal to the number of rules N. In this paper we relax that assumption and replace formula (6) by

♣ This work is supported by the Foundation for Polish Science (Professorial Grant 2005-2008), the Polish State Committee for Scientific Research (Grant Nr T11C 04827) and Special Research Grant supported by Polish Ministry of Education and Science.

Krzysztof Cpałka, Robert Nowicki and Leszek Rutkowski are with the Department of Computer Engineering at Częstochowa University of Technology, al. Armii Krajowej 36, 42-200 Częstochowa, Poland and with Department of Artificial Intelligence, Academy of Humanities and Economics, ul. Rewolucji 1905 nr 64, 90-222 Łódź, Poland (e-mail: {cpalka, rnowicki, lrutko}@kik.pcz.czest.pl,).

$$\overline{y} = \frac{\sum_{r=1}^{R} \overline{y}^r \cdot \mu_{B'}\left(\overline{y}^r\right)}{\sum_{r=1}^{R} \mu_{B'}\left(\overline{y}^r\right)} \tag{8}$$

where $R \geq 1$. For further investigations we choose neuro-fuzzy systems of a logical type with an S-implication used in formula (4) and consequently the aggregation operator, applied in order to obtain the fuzzy set B' based on fuzzy sets $\overline{B}^k$, is a t-norm. Moreover, we incorporate flexibility parameters [15], into construction new neuro-fuzzy systems. These parameters have the following interpretation:

1) weights in antecedents of the rules $w_{i,k}^{\tau} \in [0,1]$, $i = 1,\ldots,n$, $k = 1,\ldots,N$,
2) weights in aggregation of the rules $w_k^{\text{agr}} \in [0,1]$, $k = 1,\ldots,N$,
3) soft strength of firing controlled by parameter α_k^{τ}, $k = 1,\ldots,N$,
4) soft implication controlled by parameter α_k^{I}, $k = 1,\ldots,N$,
5) soft aggregation of rules controlled by parameter α^{agr}.

In view of above assumptions, we derive a flexible neuro-fuzzy system given by

$$\overline{y} = \frac{\sum_{r=1}^{R} \overline{y}^r \cdot \text{agr}_r\left(\mathbf{x}, \overline{y}^r\right)}{\sum_{r=1}^{R} \text{agr}_r\left(\mathbf{x}, \overline{y}^r\right)} \tag{9}$$

where

$$\tau_k(\overline{\mathbf{x}}) = \left(\begin{array}{l} \left(1-\alpha_k^{\tau}\right)\text{avg}\left(\mu_{A_1^k}(\overline{x}_1),\ldots,\mu_{A_n^k}(\overline{x}_n)\right) + \\ +\alpha_k^{\tau} T^{*}\left\{\begin{array}{c} \mu_{A_1^k}(\overline{x}_1),\ldots,\mu_{A_n^k}(\overline{x}_n)\ ; \\ w_{1,k}^{\tau},\ldots,w_{n,k}^{\tau} \end{array}\right\} \end{array}\right) \tag{10}$$

$$I_{k,r}\left(\overline{\mathbf{x}}, \overline{y}^r\right) = \left(\begin{array}{l} \left(1-\alpha_k^{I}\right)\text{avg}\left(1-\tau_k(\overline{\mathbf{x}}), \mu_{B^k}\left(\overline{y}^r\right)\right) + \\ +\alpha_k^{I} S\left\{1-\tau_k(\overline{\mathbf{x}}), \mu_{B^k}\left(\overline{y}^r\right)\right\} \end{array}\right) \tag{11}$$

$$\text{agr}_r\left(\overline{\mathbf{x}}, \overline{y}^r\right) = \left(\begin{array}{l} \left(1-\alpha^{\text{agr}}\right)\text{avg}\left(I_{1,r}\left(\overline{\mathbf{x}}, \overline{y}^r\right),\ldots,I_{N,r}\left(\overline{\mathbf{x}}, \overline{y}^r\right)\right) + \\ +\alpha^{\text{agr}} T^{*}\left\{\begin{array}{c} I_{1,r}\left(\overline{\mathbf{x}}, \overline{y}^r\right),\ldots,I_{N,r}\left(\overline{\mathbf{x}}, \overline{y}^r\right)\ ; \\ w_1^{\text{agr}},\ldots,w_N^{\text{agr}} \end{array}\right\} \end{array}\right) \tag{12}$$

It is easily seen that system (9)-(12) contains $N(3n+5)+R+1$ parameters to be determined in the process of learning. Using arguments similar to those in [17] the following result can be shown:

Theorem 1

The flexible neuro-fuzzy system given by formulas (9)-(12) is universal aproximator.

Now we develop an algorithm of reduction of neuro-fuzzy systems. The algorithm is based on analysis of weights in antecedents of the rules $w_{i,k}^{\tau} \in [0,1]$, $i = 1,\ldots,n$, $k = 1,\ldots,N$, and weights in aggregation of the rules $w_k^{\text{agr}} \in [0,1]$, $k = 1,\ldots,N$. The flowchart of the algorithm is depicted in Fig. 1.

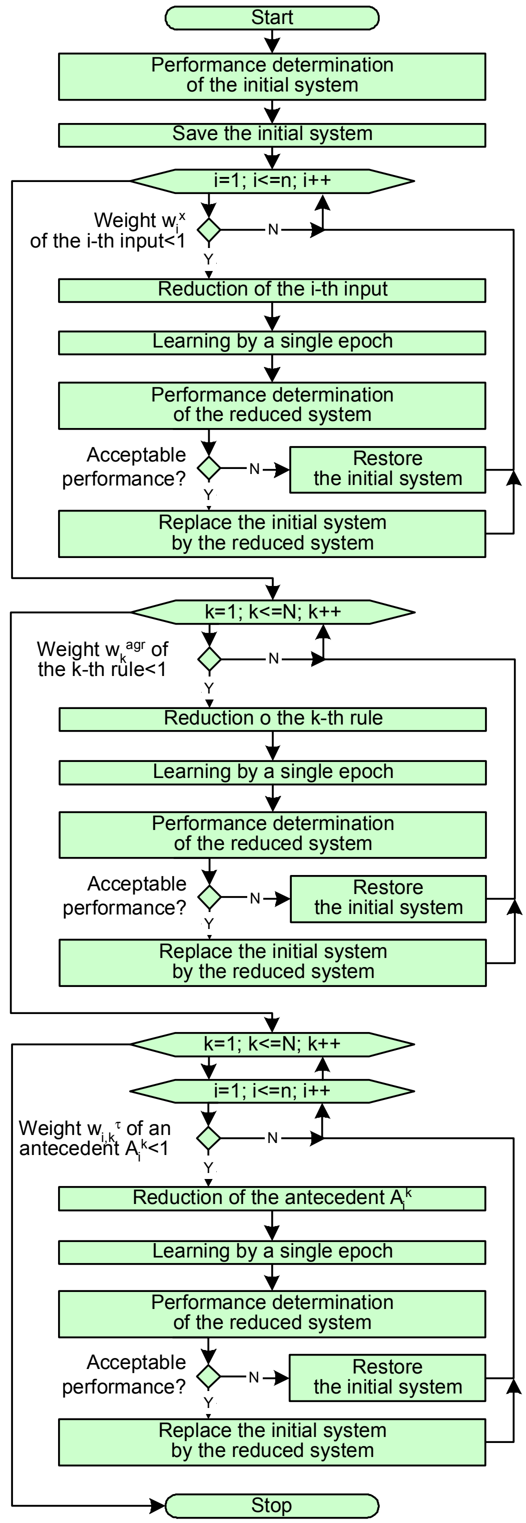

Fig. 1. The algorithm of reduction of neuro-fuzzy systems

The flowchart in Fig. 1 comprises 3-parts. First, we determine performance of the initial system (before the reduction process); for example, in a case of the classification we determine a percentage of mistakes of the system. The weights $w_i^x \in [0,1]$, $i = 1,\ldots,n$ are calculated using

$$w_i^x = \frac{1}{N} \sum_{k=1}^{N} w_{i,k}^{\tau} \tag{13}$$

In subsequent stages we reduce number of inputs, number of rules and number of antecedents.

The neuro-fuzzy system is simulated on Heart problem [16].

The Heart problem contains 270 instances and each instance is described by thirteen attributes (age, sex, chest pain type, resting blood pressure, serum cholestoral in mg/dl, fasting blood sugar > 120 mg/dl, resting electrocardiographic result, maximum heart rate achieved, exercise induced angina, oldpeak = ST depression induced by exercise relative to rest, the slope of the peak exercise ST segment, number of major vessels colored by flourosopy, tha). There are two classes: absence or presence of heart disease. In our experiments, all sets are divided into a learning sequence (189 sets) and a testing sequence (81 sets).

The experimental results for the Heart Problem problem are depicted in tables I, II, III, IV, V. In Table I we show the percentage of mistakes in the learning and testing sequences before and after reduction, e.g. for $N = 2$ and $R = 3$ we have 11.11%/11.11% for the learning sequence before and after reduction and 14.81%/14.81% for the testing sequence before and after reduction. In Table II we present number of inputs, rules, points of discretization, number of antecedents and number of parameters before and after reduction. In Table III we show degree of learning time reduction [%] and degree of learning time reduction per a single parameter [%] for a reduced system. In Table IV we present reduced inputs and antecedents. In Table V we depict percentage of neuro-fuzzy systems having a particular input (attribute) after the reduction process and percentage of inputs (attributes) corresponding to a particular neuro-fuzzy system after the reduction process.

TABLE I
SIMULATION RESULTS

HEART PROBLEM (algorithm for reduction based on analysis of weights)				
R	***N***			
	1	2	3	4
2	13.22%/13.22% 16.04%/16.04%	12.16%/12.16% 14.81%/14.81%	10.58%/10.58% 14.81%/14.81%	11.64%/11.64% 12.34%/12.34%
3	13.22%/13.22% 14.81%/14.81%	11.11%/11.11% 14.81%/14.81%	10.58%/10.58% 14.81%/14.81%	10.58%/10.58% 12.34%/12.34%
4	13.22%/13.22% 14.81%/14.81%	11.11%/11.11% 14.81%/14.81%	9.52%/9.52% 14.81%/14.81%	12.34%/12.34% 9.87%/9.87%

TABLE II
SIMULATION RESULTS

HEART PROBLEM (algorithm for reduction based on analysis of weights)				
R	***N***			
	1	2	3	4
2	13/1/2/13/47 **12**/1/2/**12**/**44**	13/2/2/26/91 **10**/2/2/**20**/**73**	13/3/2/39/135 **12**/3/2/**34**/**120**	13/4/2/52/179 **11**/4/2/**42**/**149**
3	13/1/3/13/48 **12**/1/3/**12**/**45**	13/2/3/26/92 **10**/2/3/**19**/**71**	13/3/3/39/136 **11**/3/3/**29**/**106**	13/4/3/52/180 **12**/4/3/**47**/**165**
4	13/1/4/13/49 **12**/1/4/**12**/**46**	13/2/4/26/93 **12**/2/4/**24**/**87**	13/3/4/39/137 **11**/3/4/**26**/**98**	13/4/4/52/181 13/**3**/4/**39**/**137**

TABLE III
SIMULATION RESULTS

HEART PROBLEM (algorithm for reduction based on analysis of weights)				
R	***N***			
	1	2	3	4
2	11% 5%	32% 17%	20% 10%	29% 16%
3	5% -1%	30% 11%	32% 15%	32% 27%
4	3% -2%	29% 25%	34% 10%	23% -1%

TABLE IV
SIMULATION RESULTS

HEART PROBLEM (algorithm for reduction based on analysis of weights)				
R	***N***			
	1	2	3	4
2	$\bar{x}_4$	$\bar{x}_4$, $\bar{x}_5$, $\bar{x}_8$	$\bar{x}_2$, A_4^1, A_5^1	$\bar{x}_4$, $\bar{x}_7$, A_6^1, A_8^1
3	$\bar{x}_4$	$\bar{x}_1$, $\bar{x}_2$, $\bar{x}_4$, A_{13}^1	$\bar{x}_1$, $\bar{x}_4$, A_5^1, A_6^1, A_{11}^1, A_7^2	$\bar{x}_4$, A_5^1
4	$\bar{x}_1$	$\bar{x}_4$	$\bar{x}_1$, $\bar{x}_4$, A_5^1, A_6^1, A_7^1, A_6^2, A_7^2, A_8^2, A_{13}^2	$rule_2$

TABLE V
SIMULATION RESULTS

HEART PROBLEM (algorithm for reduction based on analysis of weights)													
N R	1 2	1 3	1 4	2 2	2 3	2 4	3 2	3 3	3 4	4 2	4 3	4 4	
$\bar{x}_1$	v	v		v		v	v			v	v	v	67%
$\bar{x}_2$	v	v	v	v		v		v	v	v	v	v	83%
$\bar{x}_3$	v	v	v	v	v	v	v	v	v	v	v	v	100%
$\bar{x}_4$			v				v					v	25%
$\bar{x}_5$	v	v	v		v	v	v	v	v	v	v	v	92%
$\bar{x}_6$	v	v	v	v	v	v	v	v	v	v	v	v	100%
$\bar{x}_7$	v	v	v	v	v	v	v	v	v		v	v	92%
$\bar{x}_8$	v	v	v		v	v	v	v	v	v	v	v	92%
$\bar{x}_9$	v	v	v	v	v	v	v	v	v	v	v	v	100%
$\bar{x}_{10}$	v	v	v	v	v	v	v	v	v	v	v	v	100%
$\bar{x}_{11}$	v	v	v	v	v	v	v	v	v	v	v	v	100%
$\bar{x}_{12}$	v	v	v	v	v	v	v	v	v	v	v	v	100%
$\bar{x}_{13}$	v	v	v	v	v	v	v	v	v	v	v	v	100%
	92%	92%	92%	77%	77%	92%	92%	85%	85%	85%	92%	100%	

III. Rough-neuro-fuzzy systems

In this section we will develop a new algorithm for classification in the case of incomplete knowledge about classified object. The main idea is to combine fuzzy methods with the rough set theory. The goal of classification is to determine if the object or state x belongs to class ω_j or not. The number of classes is m and $j=1,\ldots,m$. Thus the classifier can take one of two decisions: $x\in\omega_j$ or $x\notin\omega_j$. The decision is taken based on known values of classified object features. The features can be represented as vector $\mathbf{v}_D=[v_1,v_2,\ldots,v_{n_D}]$, the values of features, as vector $\overline{\mathbf{v}}_D=[\overline{v}_1,\overline{v}_2,\ldots,\overline{v}_{n_D}]$. The n_D is the number of known feature values.

Let us assume that the neuro-fuzzy classifier comprises knowledge in the form of fuzzy rules and it works satisfactorily enough when its inputs receive the values of the attributes of all the classified objects, which were taken into account in the design process. Our goal involves developing a transformation of that classifier, so that it could also work when some data are unavailable.

In definition of the rough sets a notion of equivalence class $[x]_R$ is very important. It is defined as follows:

Definition 1

The equivalence class $[\hat{x}]_R$ is a set of elements $x\in\mathbf{X}$ which are related with object $\hat{x}$ by relation R. It is expressed as follows

$$[\hat{x}]_R=\{x\in\mathbf{X}:\hat{x}Rx\}, \tag{14}$$

where R is equivalence relation i.e. any relation satisfying reflexivity, symmetry and transitivity conditions.

Definition 2

Let us assume that A is a set defined in space $\mathbf{X}$. The rough set A is defined as a pair of sets $(\underline{R}(A),\overline{R}(A))$ where $\underline{R}(A)$ is R-lower approximation of set A and $\overline{R}(A)$ is R-upper approximation of set A. They are defined as follows [11], [12]

$$\underline{R}(A)=\{x\in\mathbf{X}:[x]_R\subseteq A\} \tag{15}$$

and

$$\overline{R}(A)=\{x\in\mathbf{X}:[x]_R\cap A\neq\varnothing\}. \tag{16}$$

The set A and its lower and upper approximations fulfill the inequality

$$\underline{R}(A)\subseteq A\subseteq\overline{R}(A) \tag{17}$$

Description of each object $x\in\mathbf{X}$ is realized through a set of n features $Q=\{v_1,v_2,\ldots,v_n\}$. The features can be also written down as a vector $\mathbf{v}=[v_1,v_2,\ldots,v_n]$. The value of the features is depicted as a vector $\overline{\mathbf{v}}=[\overline{v}_1,\overline{v}_2,\ldots,\overline{v}_n]$. The value of feature v_i of object x is expressed as value of information function family f_x

$$\overline{v}_i=f_x(v_i), \tag{18}$$

where $i=1,\ldots,n$, $v_i\in\mathbf{V}_i$ and $\mathbf{V}=\mathbf{V}_1\times\mathbf{V}_2\times\ldots\times\mathbf{V}_n$. The quadruple $SI=\{\mathbf{X},Q,\mathbf{V},f\}$ is called *information system* [11], [12].

Let us isolate the subset $D\subseteq Q$ of features. Then we can define the $\widetilde{D}$-indiscernibility relation as follows

$$x\widetilde{D}\hat{x}\Leftrightarrow\forall v_i\in D;f_x(v_i)=f_{\hat{x}}(v_i), \tag{19}$$

where $x,\hat{x}\in\mathbf{X}$.

When we apply the $\widetilde{D}$-indiscernibility relation as equivalence relation in Definitions **1** and **2**, we can define the $\widetilde{D}$-lower approximation of set A and the $\widetilde{D}$-upper approximation of set A as

$$\underline{\widetilde{D}}(A)=\{x\in\mathbf{X}:[x]_{\widetilde{D}}\subseteq A\} \tag{20}$$

and

$$\overline{\widetilde{D}}(A)=\{x\in\mathbf{X}:[x]_{\widetilde{D}}\cap A\neq\varnothing\}. \tag{21}$$

Definition 3

Fuzzy set A defined in nonempty universe $\mathbf{X}$ is a set of pairs

$$A=\{(x,\mu_A(x));x\in\mathbf{X}\}, \tag{22}$$

where

$$\mu_A:\mathbf{X}\mapsto[0,1] \tag{23}$$

is the membership function.

Referring to terms defined above in this section we can equate object x membership with its features value $\overline{\mathbf{v}}=[\overline{v}_1,\overline{v}_2,\ldots,\overline{v}_n]$ membership. So we can use interchangeably x or $\overline{\mathbf{v}}=[\overline{v}_1,\overline{v}_2,\ldots,\overline{v}_n]$ and we may as well define fuzzy set as

$$A=\{(\overline{\mathbf{v}},\mu_A(\overline{\mathbf{v}}));\overline{\mathbf{v}}\in\mathbf{V}\} \tag{24}$$

where

$$\mu_A:\mathbf{V}\mapsto[0,1] \tag{25}$$

is the membership function.

Let we note that object x is not equal to its features values $\overline{\mathbf{v}}$, however

$$\mu_A(x)=\mu_A(\overline{\mathbf{v}})=\mathop{T}_{i=1}^{n}\mu_{A_i}(\overline{v}_i), \tag{26}$$

where $A=A_1\times A_2\times\ldots\times A_n$, T is any t-norm.

Definition 4

The rough-fuzzy set is a pair $(\underline{R}A,\overline{R}A)$ of fuzzy sets. $\underline{R}A$ is a R-lower approximation and $\overline{R}A$ is a R-upper approximation of fuzzy set $A\subseteq\mathbf{X}$. The membership functions of $\underline{R}A$ and $\overline{R}A$ are defined as follows

$$\mu_{\underline{R}A}(\hat{x})=\inf_{x\in[\hat{x}]_R}\mu_A(x), \tag{27}$$

$$\mu_{\overline{R}A}(\hat{x})=\sup_{x\in[\hat{x}]_R}\mu_A(x). \tag{28}$$

For future deliberations let us consider the case when the features v_i are the real numbers.

Theorem 2

If we assume that fuzzy set A is defined by equations (26) then the membership function of its $\widetilde{D}$-lower approximation is given as follows

$$\mu_{\underline{\widetilde{D}}A}(x)=T\left\{\underset{i:v_i\in D}{T}\mu_{A_i}(\overline{v}_i),\ \underset{i:v_i\in G}{T}\inf_{v_i\in\mathbf{V}_i}\mu_{A_i}(v_i)\right\}. \tag{29}$$

The membership function of its $\widetilde{D}$-upper approximation is given by formula

$$\mu_{\overline{\widetilde{D}}A}(x)=T\left\{\underset{i:v_i\in D}{T}\mu_{A_i}(\overline{v}_i),\ \underset{i:v_i\in G}{T}\sup_{v_i\in\mathbf{V}_i}\mu_{A_i}(v_i)\right\}. \tag{30}$$

The description of the neuro-fuzzy classifier is given by

$$\overline{z}_j=\frac{\sum_{r=1}^{N}\overline{z}_j^r\mu_{A^r}(\overline{\mathbf{v}})}{\sum_{r=1}^{N}\mu_{A^r}(\overline{\mathbf{v}})}. \tag{31}$$

Obviously, equation (31) is almost identical to (6), except for variable $\overline{z}_j^r$ instead of $\overline{y}_j^r$. However, using assumption

$$\mu_{B_j^r}(z_j)=\begin{cases}1 & \text{if} \quad z_j=\overline{z}_j^r\\ 0 & \text{if} \quad z_j\neq\overline{z}_j^r\end{cases} \tag{32}$$

we can pass over the connection in structure when $\overline{z}_j^r=0$. So we obtain description of a much simpler architecture of neuro-fuzzy classifier:

$$\overline{z}_j=\frac{\sum_{\substack{r=1\\ r:\overline{z}_j^r=1}}^{N}\mu_{A^r}(\overline{\mathbf{v}})}{\sum_{r=1}^{N}\mu_{A^r}(\overline{\mathbf{v}})}. \tag{33}$$

In this section we study the neuro-fuzzy classifier in a specific situation i.e. when not complete information about object is available. Let we assume that:

a) The classifier is set up and was developed for n features of classified objects. Q denotes the set of all features of objects used in the course of system developing.

b) In the course of object x classification only values of $n_D\leq n$ features are known. $D\subseteq Q$ denotes the set of features which values are known. $G=Q\setminus D$ denotes the set of features whose values are unknown.

The classifier defined by (6) does not work in such situation. Our goal is to define the special version of neuro-fuzzy classifier which could work in the described situation. In the proposed classifier we use the rough-fuzzy set, so the system is called rough-neuro-fuzzy classifier.

It is obvious that if we assume various values of unknown features $\mathbf{v}_D$, we obtain various values of $\overline{z}_j$ as the output of neuro-fuzzy classifier. In most cases it is not possible to test all values of vector $\mathbf{v}_D$. However, it is enough to find the smallest possible value of $\overline{z}_j$ denoted as $\underline{\overline{z}}_j$ and the highest one denoted as $\overline{\overline{z}}_j$. This notation refers to notation of rough sets and rough-fuzzy sets. Value $\underline{\overline{z}}_j$ is the membership degree of the object x to $\widetilde{D}$-lower approximation of class ω_j

$$\mu_{\underline{\widetilde{D}}\omega_j}(x)=\underline{\overline{z}}_j \tag{34}$$

and $\overline{\overline{z}}_j$ is the membership degree of object x to $\widetilde{D}$-upper approximation of class ω_j

$$\mu_{\overline{\widetilde{D}}\omega_j}(x)=\overline{\overline{z}}_j \tag{35}$$

Theorem 3

Let us consider the neuro-fuzzy classifier defined by equation (6). When assumptions a) and b) are satisfied, the lower and upper approximation of the membership of object x to class ω_j is given by

$$\underline{\overline{z}}_j=\frac{\sum_{\substack{r=1\\ r:\overline{z}_j^r=1}}^{N}\mu_{A_{\mathrm{L}}^r}(\overline{\mathbf{v}})}{\sum_{r=1}^{N}\mu_{A_{\mathrm{L}}^r}(\overline{\mathbf{v}})} \tag{36}$$

and

$$\overline{\overline{z}}_j=\frac{\sum_{\substack{r=1\\ r:\overline{z}_j^r=1}}^{N}\mu_{A_{\mathrm{U}}^r}(\overline{\mathbf{v}})}{\sum_{r=1}^{N}\mu_{A_{\mathrm{U}}^r}(\overline{\mathbf{v}})}, \tag{37}$$

where A_{L}^r and A_{U}^r are defined as follows

$$A_{\mathrm{L}}^r=\begin{cases}\underline{\widetilde{D}}A^r & \text{if} \quad \overline{z}_j^r=1\\ \overline{\widetilde{D}}A^r & \text{if} \quad \overline{z}_j^r=0\end{cases} \tag{38}$$

and

$$A_{\mathrm{U}}^r=\begin{cases}\overline{\widetilde{D}}A^r & \text{if} \quad \overline{z}_j^r=1\\ \underline{\widetilde{D}}A^r & \text{if} \quad \overline{z}_j^r=0\end{cases}. \tag{39}$$

When in equations (36) and (37) we replace A_{L}^r and A_{U}^r with (38) and (39), respectively, taking into account assumption (32) we obtain a more general description of rough-neuro-fuzzy classifier:

$$\underline{\overline{z}}_j=\frac{\sum_{r=1}^{N}\overline{z}_j^r\mu_{\underline{\widetilde{D}}A^r}(\overline{\mathbf{v}})}{\sum_{r=1}^{N}\overline{z}_j^r\mu_{\underline{\widetilde{D}}A^r}(\overline{\mathbf{v}})+\sum_{r=1}^{N}\neg\overline{z}_j^r\mu_{\overline{\widetilde{D}}A^r}(\overline{\mathbf{v}})} \tag{40}$$

and

$$\overline{\overline{z}}_j=\frac{\sum_{r=1}^{N}\overline{z}_j^r\mu_{\overline{\widetilde{D}}A^r}(\overline{\mathbf{v}})}{\sum_{r=1}^{N}\overline{z}_j^r\mu_{\overline{\widetilde{D}}A^r}(\overline{\mathbf{v}})+\sum_{r=1}^{N}\neg\overline{z}_j^r\mu_{\underline{\widetilde{D}}A^r}(\overline{\mathbf{v}})}, \tag{41}$$

where $\neg\overline{z}_j^r=1-\overline{z}_j^r$.

If we need a crisp answer we should apply an appropriate defuzzification method. We suggest to use the following method

Definition 5

Let $\underline{\overline{z}}_j=\mu_{\underline{\widetilde{D}}\omega_j}(x)$ be the lower approximation of membership degree of object x to class ω_j and $\overline{\overline{z}}_j=\mu_{\overline{\widetilde{D}}\omega_j}(x)$ be its upper approximation. Let as fix two numbers (thresholds) z_{IN} and z_{OUT} such that $1 > z_{\mathrm{IN}} \geq z_{\mathrm{OUT}} > 0$. Then the crisp decision is defined as follows

$$\begin{cases} x \in \omega_j & \text{if} \quad \underline{z}_j \geq z_{\mathrm{IN}} \text{ and } \overline{z}_j > z_{\mathrm{IN}} \\ x \notin \omega_j & \text{if} \quad \underline{z}_j < z_{\mathrm{OUT}} \text{ and } \overline{z}_j \leq z_{\mathrm{OUT}} \\ \text{perhaps } x \in \omega_j & \text{if} \quad z_{\mathrm{IN}} > \underline{z}_j \geq z_{\mathrm{OUT}} \text{ and } \overline{z}_j > z_{\mathrm{IN}} \\ \text{perhaps } x \notin \omega_j & \text{if} \quad \underline{z}_j < z_{\mathrm{OUT}} \text{ and } z_{\mathrm{OUT}} < \overline{z}_j \leq z_{\mathrm{IN}} \\ \text{undefined} & \text{otherwise} \end{cases} \quad .(42)$$

When we assume that $z_{\mathrm{IN}} = z_{\mathrm{OUT}} = \frac{1}{2}$, equation (42) takes the form

$$\begin{cases} x \in \omega_j & \text{if} \quad \underline{z}_j \geq \frac{1}{2} \text{ and } \overline{z}_j > \frac{1}{2} \\ x \in \omega_j & \text{if} \quad \underline{z}_j < \frac{1}{2} \text{ and } \overline{z}_j \leq \frac{1}{2} \\ \text{undefined} & \text{otherwise} \end{cases} . \quad (43)$$

The performance of rough-neuro-fuzzy classifier (40), (41) will be tested on the Wisconsin Breast Cancer problem (WBCD). Data contain 699 instances (of which 16 instances have a single missing attribute) and each instance is described by nine attributes (clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, mitoses). We removed those 16 instances and used the remaining 683 instances. Out of 683 data samples, 444 cases represent benign breast cancer and 239 cases describe malignant breast cancer. The problem is to classify whether a new case is a benign (class 1) or malignant (class 2) type of cancer. In our experiments, all sets are divided into a learning sequence (478 sets) and a testing sequence (205 sets).

Table VI contains the eight demonstration instances patient. The first four instances come from sequence used for designing of system and the other four instances come from testing sequences. For each instance we take the value of each feature and membership of particular classes (ω_{ill} and $\omega_{\mathrm{healthy}}$). Tables VII-XI show the result of classification in case when various set of attributes is available. In Tables XII and Figure 1 we show the percentage of correct classification, no classification and incorrect classification for different sets of known features, both for sequence used for designing (d) of system and sequence used for testing (t).

IV. Final remarks

In the paper two neuro-fuzzy systems have been studied. Flexible neuro-fuzzy system presented in Section II are universal approximators. Rough-neuro-fuzzy systems derived in Section III can be applied in the case of missing data.

TABLE VI
SELECTED SAMPLES OF WBCD

x	$\overline{\mathbf{v}}$	correct conclusion
x_1	[3,1,1,1,2,1,3,1,1]	$x \notin \omega_{\mathrm{ill}}, x \in \omega_{\mathrm{healthy}}$
x_2	[8,3,3,1,2,2,3,2,1]	$x \notin \omega_{\mathrm{ill}}, x \in \omega_{\mathrm{healthy}}$
x_3	[8,9,9,5,3,5,7,7,1]	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_4	[5,3,3,3,6,10,3,1,1]	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_5	[4,3,3,1,2,1,3,3,1]	$x \notin \omega_{\mathrm{ill}}, x \in \omega_{\mathrm{healthy}}$
x_6	[6,8,8,1,3,4,3,7,1]	$x \notin \omega_{\mathrm{ill}}, x \in \omega_{\mathrm{healthy}}$
x_7	[10,4,3,1,3,3,6,5,2]	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_8	[5,3,3,1,3,3,3,3,3]	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$

TABLE VII
RESULT FOR ALL AVAILABLE FEATURES $v_1, \ldots, v_9$

x	$\underline{z}_{\mathrm{ill}}, \overline{z}_{\mathrm{ill}}$	$\underline{z}_{\mathrm{healthy}}, \overline{z}_{\mathrm{healthy}}$	conclusion
x_1	0.06, 0.06	0.94, 0.94	$x \notin \omega_{\mathrm{ill}}, x \in \omega_{\mathrm{healthy}}$
x_2	0.53, 0.53	0.47, 0.47	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_3	1.00, 1.00	0.00, 0.00	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_4	0.74, 0.74	0.26, 0.26	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_5	0.35, 0.35	0.65, 0.65	$x \notin \omega_{\mathrm{ill}}, x \in \omega_{\mathrm{healthy}}$
x_6	0.61, 0.61	0.39, 0.39	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_7	0.88, 0.88	0.12, 0.12	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_8	0.73, 0.73	0.27, 0.27	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$

TABLE VIII
RESULT FOR EIGHT AVAILABLE FEATURES $v_1, \ldots, v_7, v_9$

x	$\underline{z}_{\mathrm{ill}}, \overline{z}_{\mathrm{ill}}$	$\underline{z}_{\mathrm{healthy}}, \overline{z}_{\mathrm{healthy}}$	conclusion
x_1	0.05, 0.23	0.77, 0.95	$x \notin \omega_{\mathrm{ill}}, x \in \omega_{\mathrm{healthy}}$
x_2	0.41, 0.69	0.31, 0.59	no conclusion
x_3	1.00, 1.00	0.00, 0.00	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_4	0.59, 0.85	0.15, 0.41	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_5	0.17, 0.53	0.47, 0.83	no conclusion
x_6	0.58, 0.61	0.39, 0.42	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_7	0.86, 0.92	0.08, 0.14	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_8	0.60, 0.76	0.24, 0.40	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$

TABLE IX
RESULT FOR EIGHT AVAILABLE FEATURES $v_1, v_3, \ldots, v_9$

x	$\underline{z}_{\mathrm{ill}}, \overline{z}_{\mathrm{ill}}$	$\underline{z}_{\mathrm{healthy}}, \overline{z}_{\mathrm{healthy}}$	conclusion
x_1	0.04, 0.24	0.76, 0.96	$x \notin \omega_{\mathrm{ill}}, x \in \omega_{\mathrm{healthy}}$
x_2	0.26, 0.63	0.37, 0.74	no conclusion
x_3	0.99, 1.00	0.00, 0.01	$x \in \omega_{\mathrm{ill}}, x \notin \omega_{\mathrm{healthy}}$
x_4	0.29, 0.82	0.18, 0.71	no conclusion
x_5	0.22, 0.46	0.54, 0.78	$x \notin \omega_{\mathrm{ill}}, x \in \omega_{\mathrm{healthy}}$
x_6	0.44, 0.72	0.28, 0.56	no conclusion
x_7	0.42, 0.98	0.02, 0.58	no conclusion
x_8	0.25, 0.82	0.18, 0.75	no conclusion

TABLE X
RESULT FOR SEVEN AVAILABLE FEATURES $v_1,\dots,v_7$

x	$\underline{z}_{\text{ill}}, \overline{z}_{\text{ill}}$	$\underline{z}_{\text{healthy}}, \overline{z}_{\text{healthy}}$	conclusion
x_1	0.04, 0.29	0.71, 0.96	$x \notin \omega_{\text{ill}}, x \in \omega_{\text{healthy}}$
x_2	0.22, 0.78	0.22, 0.78	no conclusion
x_3	1.00, 1.00	0.00, 0.00	$x \in \omega_{\text{ill}}, x \notin \omega_{\text{healthy}}$
x_4	0.33, 0.93	0.07, 0.67	no conclusion
x_5	0.12, 0.61	0.39, 0.88	no conclusion
x_6	0.53, 0.77	0.23, 0.47	$x \in \omega_{\text{ill}}, x \notin \omega_{\text{healthy}}$
x_7	0.61, 1.00	0.00, 0.39	$x \in \omega_{\text{ill}}, x \notin \omega_{\text{healthy}}$
x_8	0.23, 0.86	0.14, 0.77	no conclusion

TABLE XI
RESULT FOR FIVE AVAILABLE FEATURES $v_1, v_6,\dots,v_9$

x	$\underline{z}_{\text{ill}}, \overline{z}_{\text{ill}}$	$\underline{z}_{\text{healthy}}, \overline{z}_{\text{healthy}}$	conclusion
x_1	0.01, 1.00	0.00, 0.99	no conclusion
x_2	0.05, 1.00	0.00, 0.95	no conclusion
x_3	0.82, 1.00	0.00, 0.18	$x \in \omega_{\text{ill}}, x \notin \omega_{\text{healthy}}$
x_4	0.02, 0.97	0.03, 0.98	no conclusion
x_5	0.01, 0.99	0.01, 0.99	no conclusion
x_6	0.05, 0.98	0.02, 0.95	no conclusion
x_7	0.03, 1.00	0.00, 0.97	no conclusion
x_8	0.02, 0.98	0.02, 0.98	no conclusion

TABLE XII
PERFORMANCE OF CLASSIFIER FOR WBCD PROBLEM

available features	correct class. [%]	no class. [%]	incorect class. [%]
$v_1,\dots,v_9$	97.07/98.54	0.00/0.00	2.93/1.46
$v_1,\dots,v_8$	88.46/90.51	10.26/8.76	1.28/0.73
$v_1,\dots,v_7,v_9$	86.08/88.32	13.28/11.31	0.64/0.36
$v_1,\dots,v_6,v_8,v_9$	88.92/90.88	10.71/9.12	0.37/0.00
$v_1,\dots v_5,v_7,\dots,v_9$	85.62/86.50	13.92/13.14	0.46/0.36
$v_1,\dots v_4,v_6,\dots,v_9$	83.06/85.77	16.12/13.14	0.82/1.09
$v_1,\dots v_3,v_5,\dots,v_9$	58.70/57.30	40.75/42.70	0.55/0.00
$v_1,v_2,v_4,\dots,v_9$	86.54/90.88	12.00/9.12	1.47/0.00
$v_1,v_3,\dots,v_9$	90.20/90.15	8.61/8.76	1.19/1.09
$v_2,\dots,v_9$	92.77/91.61	6.32/7.66	0.92/0.73
$v_1,\dots,v_7$	56.96/59.85	42.86/40.15	0.18/0.00
$v_1,\dots,v_6,v_9$	54.67/60.95	45.33/39.05	0.00/0.00
$v_1,\dots,v_5,v_8,v_9$	19.69/19.34	80.31/80.66	0.00/0.00
$v_1,\dots,v_4,v_7,\dots,v_9$	32.78/32.12	66.85/67.88	0.37/0.00
$v_1,\dots,v_3,v_6,\dots,v_9$	32.88/37.59	67.12/62.41	0.00/0.00
$v_1,v_2,v_5,\dots,v_9$	27.38/30.29	72.44/69.71	0.18/0.00
$v_1,v_4,\dots,v_9$	59.62/66.79	40.02/33.21	0.37/0.00
$v_3,\dots,v_9$	80.86/82.12	18.68/17.52	0.46/0.36
$v_1,\dots,v_6$	12.45/11.68	87.55/88.32	0.00/0.00
$v_1,\dots,v_5,v_9$	10.71/10.58	89.29/89.42	0.00/0.00
$v_1,\dots,v_4,v_8,v_9$	15.84/14.23	84.16/85.77	0.00/0.00
$v_1,\dots,v_3,v_7,\dots,v_9$	0.00/0.00	100.00/100.00	0.00/0.00
$v_1,v_2,v_6,\dots,v_9$	0.18/0.73	99.82/99.27	0.00/0.00
$v_1,v_5,\dots,v_9$	13.55/14.60	86.45/85.40	0.00/0.00
$v_4,\dots,v_9$	38.28/43.07	61.54/56.93	0.18/0.00
$v_1,\dots v_5$	0.00/0.00	100.00/100.00	0.00/0.00
$v_1,\dots,v_4,v_9$	8.70/7.66	91.30/92.34	0.00/0.00
$v_1,\dots,v_3,v_8,v_9$	0.00/0.00	100.00/100.00	0.00/0.00
$v_1,v_2,v_7,\dots,v_9$	0.00/0.00	100.00/100.00	0.00/0.00
$v_1,v_6,\dots,v_9$	0.00/0.00	100.00/100.00	0.00/0.00
$v_5,\dots,v_9$	5.68/6.57	94.32/93.43	0.00/0.00
...	...	...	...
none	0.00/0.00	100.00/100.00	0.00/0.00

REFERENCES

[1] D. Dubois, H. Prade, "Rough fuzzy sets and fuzzy rough sets," Internat. J. General Systems 17 (2-3), pp. 191-209, 1990.

[2] D. Dubois, H. Prade, "Putting rough sets and fuzzy sets together," R. Słowiński (ed.) Intelligent Decision Support: Handbook of Applications and Advences of the Rough Sets Theory, Kluwer, Dordrecht 1992, pp. 203-232.

[3] C. Fodor, On fuzzy implication, *Fuzzy Sets and Systems*, vol. 42, pp. 293-300, 1991.

[4] M. Gorzałczany, *Computational Intelligence Systems and Applications: Neuro-Fuzzy and Fuzzy Neural Synergisms*. Springer-Verlag 2002.

[5] L. I. Kuncheva. *Fuzzy Classifier Design*, Physica-Verlag, New York, 2000.

[6] D. Nauck, F. Klawonn, R. Kruse, *Foundations of Neuro-Fuzzy Systems*, Wiley, 1997.

[7] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*, Prentice Hall PTR, Upper Saddle River, NJ 2001.

[8] R. Nowicki, "Rough Sets in the Neuro-Fuzzy Architectures Based on Monotonic Fuzzy Implications," *Lecture Notes in Artificial Intelligence* 3070, Artificial Intelligence and Soft Computing - ICAISC 2004, L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L. A. Zadeh (Eds.), Springer-Verlag Berlin Heidelberg, pp.~510-517, 2004.

[9] R. Nowicki, "Rough Sets in the Neuro-Fuzzy Architectures Based on Non-monotonic Fuzzy Implications," *Lecture Notes in Artificial Intelligence* 3070, Artificial Intelligence and Soft Computing - ICAISC 2004, L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L. A. Zadeh (Eds.), Springer-Verlag Berlin Heidelberg, pp.~518-525, 2004.

[10] R. Nowicki, “Rough-Neuro-Fuzzy System with MICOG Defuzzification”, 2006 Internetional Join Conference on /Neural Networks, IEEE World Congress on Computational Intelligence, Vancouver, BC, Canada 2006.

[11] Z. Pawlak, "Rough sets," *International Journal of Information and Computer Science*, Vol. 11, No. 341, pp.~341-356 , 1982.

[12] Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning About Data*, Dordrecht, Kluwer 1991.

[13] L. Rutkowski, *Flexible Neuro-Fuzzy Systems*. Kluwer Academic Publishers, 2004.

[14] L. Rutkowski and K. Cpałka, Designing and learning of adjustable quasi triangular norms with applications to neuro fuzzy systems, *IEEE Trans. on Fuzzy Systems*, vol. 13, Feb 2005, pp. 140 151.

[15] L. Rutkowski, K. Cpałka, Flexible neuro-fuzzy systems, IEEE Transactions on Neural Networks, vol. 14, May 2003, pp. 554 574.

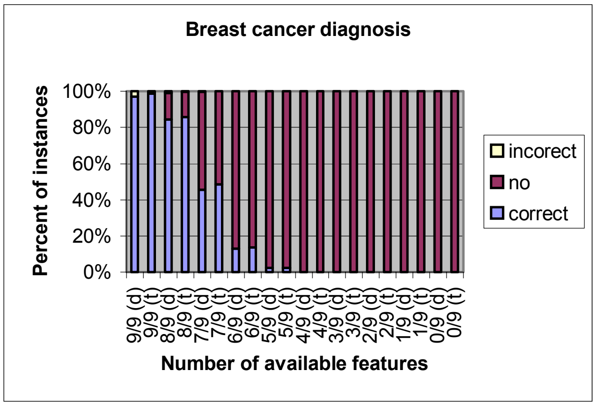

Fig. 2. The average percentage participation of classification in dependence of number of available features

[16] UCI respository of machine learning databases, Available online: *http://ftp.ics.uci.edu/pub/machine-learning-databases/*.

[17] L. X. Wang, *Adaptive Fuzzy Systems and Control*, PTR Prentice Hall, Englewood Cliffs 1994.

[18] R. R. Yager, D. P. Filev, *Essentials of fuzzy modelling and control*. John Wiley & Sons, 1994.