

# Ranked Rules and Data Visualization<sup>1</sup>

Leon Bobrowski<sup>2,3</sup>, Tomasz Sowiński<sup>2</sup>

<sup>2</sup> Institute of Computer Science of Technical University of Białystok, Poland

<sup>3</sup> Institute of Biocybernetics and Biomedical Engineering PAS, Warsaw

**Abstract.** The design of non-linear visualizing transformations of data sets that allow for a good separation of classes (categories) on the diagnostic maps is considered. The proposed transformations are based on a model of the ranked family of decision rules. Such family of rules could be generated by separate and conquer algorithms.

**Key words.** Extraction of rules from data, ranked decision rules, visualization of multivariate data, diagnostic maps

## 1 Introduction

Knowledge discovered in databases is usually represented in the form of logical rules which can be used, for example, in expert systems [1]. A chain of rules is a convenient manner of explanation and justification of decisions suggested by a computer system. The graphical representation of the information contained in a database is also important because it allows for deeper involvement of users in a decision making process.

Pattern recognition methods could be often adopted to solving the data mining problems [2]. Neural networks models can be also used in the design of computational tools for data mining and knowledge discovery in databases ([3], [4]). We are referring here to the so-called neural ranked networks [5]. The ranked layers could be generated in accordance with separate-and-conquer or covering strategy ([6], [7]). Such layers may be used in the design of non-linear visualizing transformations of data sets in the form of the diagnostic maps [8].

The diagnostic maps constitute a graphical representation of data, which results from the compression of data sets which greatly preserves their separability. The selected groups of patients (classes, categories) could be well separated on the map.

---

<sup>1</sup> This work was partially supported by the grant 8T11E00811 from the State Committee for Scientific Research (KBN) in Poland, and by the grant W/II/1/97 from Technical University of Białystok

The diagnostic maps constitute the main tool of diagnosis support in the computer system "Hepar" ([9], [10]). In this paper, the design of the visualizing transformations based on a family of logical rules is described.

## 2 The learning sets and the ranked decision rules

Let us assume that we have access to a database with descriptions of some objects  $O^j$  ( $j=1, \dots, m$ ). In result of some measurements, a set of  $n$  discrete or continuous numbers (*features*)  $x_i$  could be related to the each object  $O^j$  ( $x_i \in \{0,1\}$  or  $x_i \in \mathbb{R}^1$ ). The data describing objects  $O^j$  are represented in the form of column *feature vectors*  $\mathbf{x}^j = [x_{j1}, \dots, x_{jN}]^T$  belonging to  $N$ -dimensional *feature space*. We assume that the objects' descriptions in the database are *labeled*, which means that each object  $O^j$  is related to one of the *classes* (category)  $\omega_k$  ( $k = 1, \dots, K$ ). The  $k$ -th learning set  $C_k$  contains  $m_k$  feature vectors  $\mathbf{x}^j(k)$  associated with the class  $\omega_k$ :

$$C_k = \{ \mathbf{x}^j(k) \} \quad (j = 1, \dots, m_k). \quad (1)$$

The learning sets are *separable* if they are disjointed in the given feature space:

$$(\forall k' \neq k) \quad C_{k'} \cap C_k = \emptyset. \quad (2)$$

This means, that the elements  $\mathbf{x}^j(k')$  and  $\mathbf{x}^j(k)$  are always different if they belong to two different learning sets  $C_k$ .

In the framework of pattern recognition, the learning sets  $C_k$  are used to solve the classification problem. During the classification process, the feature vectors  $\mathbf{x}$  of unknown origin are allocated into one of the classes  $\omega_k$  in accordance with an *allocation rule*  $d_{\mathbf{w}}(\mathbf{x})$  which may depend on some vector of parameters  $\mathbf{w}$  ( $\mathbf{w} \in \mathbb{R}^N$ ):

$$\{ d_{\mathbf{w}}(\mathbf{x}) = k \} \Rightarrow \{ \mathbf{x} \text{ is allocated into class } \omega_k \}. \quad (3)$$

The choice of the vector  $\mathbf{w} = [w_1, \dots, w_N]^T$  could be made on the basis of the learning sets  $C_k$  in accordance with the *reclassification postulate*:  
*The parameters  $\mathbf{w}$  (3) should be chosen in such a manner that each feature vector  $\mathbf{x}$  from the learning sets is properly classified:*

$$(\forall k \in \{1, \dots, K\}) \quad (\forall \mathbf{x} \in C_k) \quad d_{\mathbf{w}}(\mathbf{x}) = k. \quad (4)$$

The above demand characterizes the *deterministic approach* to the design of the classifiers. In a *statistical approach* we are satisfied if only a *dominant*  $(1-\alpha)$ -part of the feature vectors  $\mathbf{x}^j$  from the learning sets is properly classified.

The allocation rule  $d_{\mathbf{v}}(\mathbf{x})$  could be based on the layer of  $n'$  *formal neurons*  $FN_i$  described by the following decision rule  $r_i'(\mathbf{w}, \theta; \mathbf{x})$ :

$$r_i' = r_i'(\mathbf{w}, \theta; \mathbf{x}) = 1 \text{ if } \langle \mathbf{w}, \mathbf{x} \rangle \geq \theta. \quad (5)$$

$$r_i' = r_i'(\mathbf{w}, \theta; \mathbf{x}) = 0 \text{ if } \langle \mathbf{w}, \mathbf{x} \rangle < \theta.$$

where  $\mathbf{w} = [w_1, \dots, w_N]^T \in \mathbb{R}^N$  is the weight vector,  $\theta$  is the threshold, and  $\langle \mathbf{w}, \mathbf{x} \rangle = \sum w_i x_i$  is the inner product.

The formal neuron  $FN_i$  is excited ( $r_i' = 1$ ) if and only if the weighted sum of inputs  $x_i$  is greater than the threshold  $\theta$ .

Let us consider also the layer build from the  $n'$  logical elements  $E_i$  operating in a parallel manner on the same feature vectors  $\mathbf{x}$  in accordance with the following decision rule:

$$r_i = r_i(\mathbf{w}, \theta; \mathbf{x}) = 1 \text{ if } w_i x_i \geq \theta. \quad (6)$$

$$r_i = r_i(\mathbf{w}, \theta; \mathbf{x}) = 0 \text{ if } w_i x_i < \theta.$$

The above decision rule could be treated as a special case of the formal neuron rule (5) which depends on only two parameters  $w_i$  and  $\theta$ .

The element  $E_i$  partitions the feature space  $\mathbf{X}$  into two regions separated by the following hyperplane:

$$H_i(\mathbf{w}, \theta; \mathbf{x}) = \{ \mathbf{x}: w_i x_i = \theta \}. \quad (7)$$

The hyperplane  $H_i(\mathbf{w}, \theta; \mathbf{x})$  is parallel to all but the  $i$ -th coordinate axis of the feature space  $\mathbf{X}$ . The feature vector  $\mathbf{x}$  excites the element  $E_i$  ( $r_i = 1$ ) if and only if  $\mathbf{x}$  is located on the positive side ( $w_i x_i > \theta$ ) of the hyperplane  $H_i(\mathbf{w}, \theta; \mathbf{x})$ .

The allocation rule  $d_v(\mathbf{r}(\mathbf{x}))$  (3) of the classifier based on the layer of  $n'$  logical elements  $E_i$  is determined by the output vector  $\mathbf{r} = [r_1, \dots, r_n]^T$  of this layer. In a particular case, the ranked allocation rule could be used.

*Definition 1:* The ranked allocation rule has the following general structure:

$$\begin{aligned} &\text{if } ((r_1(\mathbf{x})=0) \wedge (r_2(\mathbf{x})=0) \wedge \dots \wedge (r_l(\mathbf{x})=1)) \\ &\text{then } (\mathbf{x} \text{ is allocated into the class } \omega_{k(l)}) \end{aligned} \quad (8)$$

where the number  $k(l)$  of the chosen class  $\omega_{k(l)}$  is determined by the assigned number ("rank")  $l$  of the first excited element  $E_i$  in the layer ( $l \leq n'$ ).

*Definition 2:* The layer of  $n'$  elements  $E_i$  (formal neurons  $FN_i$ ) is ranked with respect to the learning sets  $C_k$  if and only if there exists a decision rule (8) which allows for correct classification of the all elements  $\mathbf{x}^j(k)$  of these sets.

The elements constituting the ranked layer are indexed (ordered) in such a manner that the elements  $E_i$  (or their decisions rules  $r_i(\mathbf{x})$ ) with low values of the index  $i$  are the most "important". For example, the rule  $r_2(\mathbf{x})$  is more important than  $r_3(\mathbf{x})$ . In accordance with (8), the allocation is based on the most important, excited element  $E_i$ .

The order among the elements  $E_i$  results from described later the separate-and-conquer procedure of the layer design. Let us define, for the moment, the support set  $A_i$  of the decision rule  $r_i(\mathbf{x})$ , as the set of such feature vectors  $\mathbf{x}$  which excite the element  $E_i$  ( $r_i(\mathbf{x}) = 1$ ). We can remark that if the support sets  $A_i$  are disjoint ( $A_i \cap A_j = \emptyset$ ), then the elements  $E_i$  could be ranked in an arbitrary manner.

*Definition 3:* The layer of  $n'$  elements  $E_i$  is  $\alpha$ -ranked with respect to the learning sets  $C_k$  if and only if there exists a decision rule (8) which allows for correct classification of at least the dominant  $(1-\alpha)$ -part of the all elements  $\mathbf{x}^j(k)$  of these sets.

The last definition will be used in a statistical approach to classification, when we are satisfied with only a dominant part of the feature vectors  $\mathbf{x}^j(k)$  from the learning sets being properly classified.

The procedure of designing a layer of formal neurons  $FN_i$  which is ranked with respect to the learning sets  $C_k$  has been described in the papers [8] and [12]. This multistage procedure could be used also in design of the ranked layer from the logical elements  $E_i$ . The procedure is based on finding the sequence of so called *admissible* hyperplanes  $H_i(\mathbf{w}^*, \theta^*; \mathbf{x})$  (7).

*Definition 4:* The hyperplanes  $H_i(\mathbf{w}^*, \theta^*; \mathbf{x})$  are *admissible* with respect to the family of learning sets  $\{C_k\}$  if and only if the vectors  $\mathbf{x}^j(k)$  from only one set  $C_k$  are situated on its positive side. This means that:

$$(\exists k) (\mathbf{w}_i^* \mathbf{x}_{ji} \geq \theta^*) \Rightarrow (\mathbf{x}^j \in C_k). \quad (9)$$

*Definition 5:* The hyperplanes  $H_i(\mathbf{w}^*, \theta^*; \mathbf{x})$  are  $\alpha$ -*admissible* with respect to the family of the learning sets  $\{C_k\}$  if and only if at least the  $(1-\alpha)$ -part of the vectors  $\mathbf{x}^j(k)$  situated on its positive side belongs to only one learning set  $C_k$ .

The vectors  $\mathbf{x}^j(k)$  which are situated on the positive side of the admissible ( $\alpha$ -admissible) hyperplane  $H_i(\mathbf{w}^*, \theta^*; \mathbf{x})$  (7) are removed before the next stage of the design procedure. In other words, the learning sets  $C_k$  are reduced during successive  $n$  stages of the design procedure. This procedure could be treated as a special case of the separate-and-conquer strategy [2]. Generally, the separate-and-conquer algorithm searches for a rule that explains a part of the learning sets  $C_k$ , separates these examples and recursively conquers the remaining examples by designing more rules until no examples remain.

The separate-and-conquer procedure allows us to find the sequence of admissible or  $\alpha$ -admissible hyperplanes  $H_i(\mathbf{w}, \theta; \mathbf{x})$  (7). As a result, the ranked allocation rule (8) can be established. Note that this rule could be represented in the following form:

$$\begin{aligned} \text{if } ((\gamma_{i(1)} \mathbf{x}_{i(1)} < \theta_{i(1)}) \wedge (\gamma_{i(2)} \mathbf{x}_{i(2)} < \theta_{i(2)}) \wedge \dots \wedge (\gamma_{i(l)} \mathbf{x}_{i(l)} \geq \theta_{i(l)}) \\ \text{then } (\mathbf{x} \text{ is allocated into the class } \omega_{k(1)}) \end{aligned} \quad (10)$$

where  $\gamma_i \in \{1, -1\}$  and  $\theta_i$  is the threshold value of the  $i$ -th element  $E_i$ .

Each of the above subrules or hyperplanes  $H_i(\gamma_i, \theta_i; \mathbf{x})$  (7) is defined by two parameters  $\gamma_m$  and  $\theta_m$ . These parameters are to be found in accordance with the postulate of *admissibility* or  $\alpha$ -*admissibility* with respect to a given family of the learning sets  $\{C_k\}$ . An  $\alpha$ -*admissibility* is taken into consideration in the case of the statistical approach to design of the ranked layers. From a computational point of view it is relatively easy to find the numbers  $\gamma_m$  and  $\theta_m$  using a discrete, exhaustive search. The basis exchange algorithms, similar to linear programming methods, have been developed for the purpose of finding the admissible hyperplanes related to the formal neurons  $FN_i$  [12]. These algorithms allow to find the solution in efficient manner even in the case of large learning sets  $C_k$ .

### 3 Data transformations based on the ranked layer

The feature vectors  $\mathbf{x}^j$  from the learning sets  $C_k$  are transformed by the ranked layer built from  $n$  elements  $E_i$  into  $n$ -dimensional vectors  $\mathbf{r}^l$ , with binary components  $r^l_i$ . This transformation has the following general form:

$$\mathbf{r}^l = F(\mathbf{W}; \mathbf{r}) . \quad (11)$$

where  $\mathbf{W}$  is the vector of the layer parameter.

A few feature vectors  $\mathbf{x}^j$  could be integrated into one vector  $\mathbf{r}^l$  in accordance with the transformations (15) (we can enumerate the vectors  $\mathbf{r}^l$  in such a manner that they are different:  $l \neq l' \Rightarrow \mathbf{r}^l \neq \mathbf{r}^{l'}$ ). As a result, the data sets are compressed and the number  $m'$  of the different vectors  $\mathbf{r}^l$  could be significantly less than  $m$ . Let us define the *active fields*  $S_1(\mathbf{W})$  of the layer with the transformation function  $F(\mathbf{W}; \mathbf{x})$ :

$$S_1 = S(\mathbf{W}; \mathbf{r}^l) = \{ \mathbf{x}^j : F(\mathbf{W}; \mathbf{x}^j) = \mathbf{r}^l \text{ and } (\exists C_k: \mathbf{x}^j \in C_k) \} . \quad (12)$$

The feature vectors  $\mathbf{x}^j$  belonging to one active field  $S_1$  are integrated into one vector  $\mathbf{r}^l$  by the ranked layer.

*Definition 6:* The active field  $S_1$  is of the *clear* type if and only if it contains the feature vectors  $\mathbf{x}^j$  from one learning set  $C_k$ . Equivalently, the *mixed* active field contains the elements of more than one learning set  $C_k$  (1).

*Definition 7:* The learning set  $C_k$  is *dominant* in the active field  $S_1$  if most of the feature vectors  $\mathbf{x}^j$  constituting  $S_1$  belong to  $C_k$ . Let symbol  $S_1(k)$  mean the active field with the dominant set  $C_k$ , or in other words, the active field related to the class  $\omega_k$ .

The class membership of the vectors  $\mathbf{r}^l$  could be defined (*assigned*) in the following manner:

$$\{ \text{the learning set } C_k \text{ is dominant in the active field } S_1 \} \Rightarrow \{ \mathbf{r}^l \in \omega_k \} \quad (13)$$

The labeled vectors  $\mathbf{r}^l$  constitute the transformed learning sets  $D_k$ :

$$D_k = \{ \mathbf{r}^l(k) \} (l = 1, \dots, m'_k) . \quad (14)$$

where  $m'_k$  is the number of transformed vectors  $\mathbf{r}^l(k)$  belonging to the class  $\omega_k$  ( $m'_k \leq m_k$ ).

The most important property of the ranked layer is its (*separable*) *linearization* of the learning sets ([11], [12]). This means that the sets  $D_k$  of the vectors  $\mathbf{r}^l$  transformed by the ranked layer are linearly separable:

$$\begin{aligned} (\forall k = 1, \dots, K) (\exists \mathbf{w}_k, \theta_k) (\forall \mathbf{r}^l \in D_k) \langle \mathbf{w}_k, \mathbf{r}^l \rangle \geq \theta_k \\ \text{and } (\forall \mathbf{r}^l \in D_i, i \neq k) \langle \mathbf{w}_k, \mathbf{r}^l \rangle < \theta_k \end{aligned} \quad (15)$$

The family of the sets  $D_k$  is *linearly separable* if each set  $D_k$  can be separated in the feature space from the sum of all remaining sets  $D_i$  by some hyperplane  $H'(\mathbf{w}, \theta; \mathbf{x})$  (9). This linearization property could be used, among other applications, in the design of the visualizing transformations.

## 4 Design of the ranked visualizing transformations

We consider special nonlinear transformations from the  $N$ -dimensional feature space  $\mathbf{X}$  onto a plane (two dimensions):

$$y_1 = F_1(\mathbf{W}_1; \mathbf{x}) \quad (16)$$

$$y_2 = F_2(\mathbf{W}_2; \mathbf{x})$$

where  $\mathbf{y} = [y_1, y_2]^T$  and  $\mathbf{W}_k$  ( $k = 1, 2$ ) are parameters of the transformations. Each of the above two transformations is related to one axis of the plane and will be based on a separate layer of the elements  $E_i$ , ranked with respect to some sets  $G_j$  as described below.

The feature vectors  $\mathbf{x}^j(k)$  are transformed and visualized as the points  $\mathbf{y}^l(k)$  on the plane. The vectors  $\mathbf{y}^l(k)$  belonging to one class  $\omega_k$  constitute the transformed learning sets  $F_k$  (14).

$$F_k = \{ \mathbf{y}^l(k) \} (l = 1, \dots, m'_k) \quad (17)$$

The so called *diagnostic maps* of the learning sets  $C_k$  can be generated by the transformations (16). A diagnostic map visualizes the data in such a manner that the selected sets  $F_k$  are well separated. In other words, the feature vectors  $\mathbf{x}^j(k)$  belonging to the selected sets  $C_k$  are transformed and visualized as the points  $\mathbf{y}^l(k)$  located in the separate regions of the map. A decision support rule could be based on the diagnostic map in the following manner:

**if** the transformed feature vector  $\mathbf{y}(\mathbf{x})$  (16) is located on the map (18)  
in the region of the well separated set(-s)  $F_k$   
**then** the vector  $\mathbf{x}$  should be allocated into the class(-es)  $\omega_k$ .

Let us divide the data set twice into some subsets  $G_{k1}$  and  $G_{k2}$  ( $k = 1, 2$ ) of the feature vectors  $\mathbf{x}^j$  in order to design the transformations (16). Two ranked layers of the elements  $E_i$ , are designed by using the subsets  $G_{ki}$ . The first layer is ranked with respect to the sets  $\{G_{11}, G_{12}\}$  ( $G_{11}$  vs  $G_{12}$ ), and the second layer is ranked with respect to the sets  $\{G_{21}, G_{22}\}$ . The sets  $G_{k1}$  and  $G_{k2}$  will be transformed by the  $k$ -th ranked layer into the sets  $D_{k1}$  and  $D_{k2}$  of the vectors  $\mathbf{r}^j$ . In accordance with the previous consideration, the sets  $D_{k1}$  and  $D_{k2}$  are linearly separable (19). Let  $(\mathbf{v}_k^*, \theta_k^*)$  be the parameters of a hyperplane which separates the sets  $D_{k1}$  and  $D_{k2}$ . These parameters can be used also in definition of the linear transformation of the transformed vectors  $\mathbf{r}_k^j$  from the sets  $D_{k1}$ :

$$\begin{aligned} y_{1j} &= \langle \mathbf{v}_1^*, \mathbf{r}_1^j \rangle + \theta_1^* \\ y_{2j} &= \langle \mathbf{v}_2^*, \mathbf{r}_2^j \rangle + \theta_2^* \end{aligned} \quad (19)$$

The transformed vectors  $\mathbf{y}^j = [y_{j1}, y_{j2}]^T$  will constitute points on the diagnostic maps which represent the feature vectors  $\mathbf{x}^j$ .

In summary, the transformation of the feature vectors  $\mathbf{x}^j$  into the points  $\mathbf{y}^j$  on the diagnostic maps is done in two steps. During the first step, each feature vector  $\mathbf{x}^j$  is transformed by two parallel ranked layers into two vectors  $\mathbf{r}_1^j$  and  $\mathbf{r}_2^j$ . During the second step, the vectors  $\mathbf{r}_1^j$  and  $\mathbf{r}_2^j$  are transformed linearly (19) into the point  $\mathbf{y}^j$ .

Note that the diagnostic map designed in this way is divided into the following quarters:

$$\begin{aligned} \text{the upper-right quarter} &- G_{11} \cap G_{21} \\ \text{the lower-right quarter} &- G_{11} \cap G_{22} \\ \text{the lower-left quarter} &- G_{12} \cap G_{22} \end{aligned} \quad (20)$$

the upper-left quarter -  $G_{12} \cap G_{21}$

The *Iris* data sets have been visualized in this manner (Fig.1).

## 5 An example of the diagnostic map design

The Fisher's *Iris* data sets contain 150 four-dimensional feature vectors  $x^j$  from three classes  $\omega_k$  (learning sets  $C_k$ ):  $\omega_1$  - *Iris setosa*,  $\omega_2$  - *Iris versicolor*,  $\omega_3$  - *Iris virginica* [3]. Each learning set  $C_k$  contains in this case exactly 50 feature vectors  $x^j$ . The transformation (16) related to the first axis of the diagnostic map has been defined by the layer of 7 elements  $E_i$  with the rule (14) ranked with respect to two sets  $C_2$  and  $C_1 \cup C_3$  (partition  $C_2$  vs.  $C_1 \cup C_3$ ) (Tab.1). Similarly, the layer of the second axis was ranked with respect to sets  $C_2$  and  $C_1 \cup C_3$  (Tab.2).

Table 1. Ranked layer designed for the partition  $C_2$  vs.  $C_1 \cup C_3$  (first axis)

Rule index	Index of control feature $x_i$	Threshold $\theta_i$	Direction $\gamma_i$	Dominant Classes $\omega_i$	Numbers of the separated feature vectors
1	3	2.45	-1	$\omega_1 \cup \omega_3$	50 (34 %)
2	4	1.8	1	$\omega_1 \cup \omega_3$	43 (29 %)
3	3	4.5	-1	$\omega_2$	36 (24 %)
4	2	3.05	1	$\omega_2$	5 (3 %)
5	3	5.1	1	$\omega_1 \cup \omega_3$	5 (3 %)
6	1	6	1	$\omega_2$	9 (6 %)
7	1	6	-1	$\omega_1 \cup \omega_3$	2 (1 %)

Table 2. Ranked layer designed for the partition  $C_3$  vs.  $C_1 \cup C_2$  (second axis)

Rule index	Index of control feature $x_i$	Threshold $\theta_i$	Direction $\gamma_i$	Dominant Classes $\omega_i$	Numbers of the separated feature vectors
1	4	1.4	-1	$\omega_1 \cup \omega_2$	84 (56 %)
2	4	1.8	1	$\omega_3$	40 (27 %)
3	3	5.1	1	$\omega_3$	7 (5 %)
4	1	6	1	$\omega_1 \cup \omega_2$	11 (7 %)
5	2	3	1	$\omega_1 \cup \omega_2$	4 (3 %)
6	2	3	-1	$\omega_3$	3 (2 %)
7	1	3	1	$\omega_1 \cup \omega_2$	1 (< 1 %)

The linear transformations (19) related to each of the two axis has been defined by the following parameters:

$$v_1^* = [v_1, \dots, v_7]^T = [-10, -2, 6, 2, -2, -10, -10]^T, \theta_1^* = 11$$

$$v_2^* = [v_1, \dots, v_7]^T = [-10, 2, 2, 0, -10, -10, 0]^T, \theta_2^* = 9$$

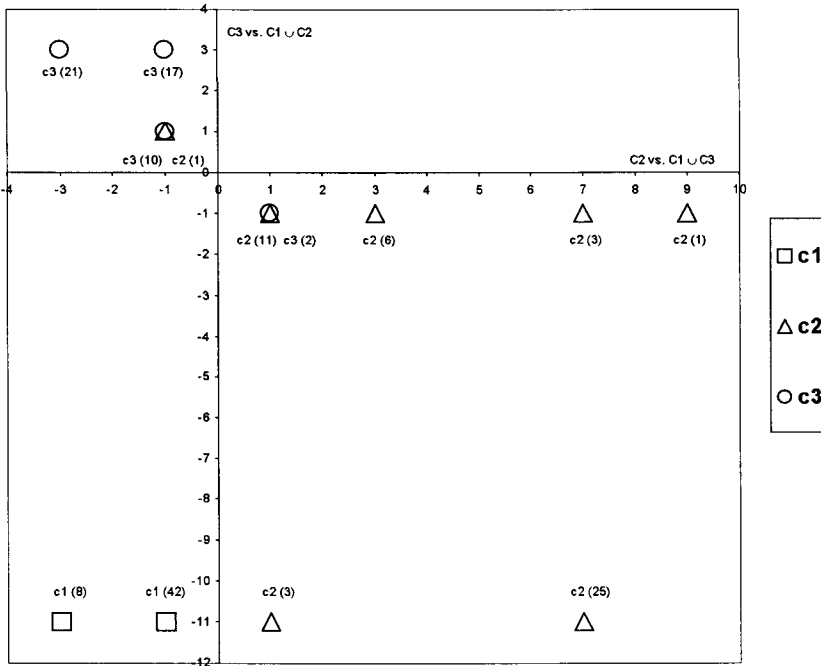


Fig. 1. Diagnostic map for Iris data

The characteristic feature of the map is that one set  $C_k$  of the objects  $x^j$  is located in the one quarter of the visualising plane. Any four separable data sets could be transformed in this way on the plane

## 6 Concluding remarks

The procedure for designing the visualising transformations has been described here. These non-linear transformations can be based on the ranked layer of the decision rules which are extracted from the learning data sets. The visualising transformation defines a new structure (topology) of the feature space  $X$ .

The described procedure can be applied to the design of diagnostic maps, which can be used in the decision support (18) to complement logical formulas. The maps make possible the interactive user involvement in the decision making process.



Another possible application of the maps is a search for (non-linear) interactions among a groups of objects. The visualizing transformation designed on the base of the selected four categories (groups) of objects could be used to visualize the same or another object divided into other categories. We could observe in which way the selected partitions of the feature space  $X$  interact.

One of the problems which should be avoided during the design of the non-linear visualizing transformations is data overfitting. The transformation based on ranked networks may depend on a large number of parameters which have to be estimated from the learning data sets. The large number of estimated parameters could diminish the generality of the design transformations and the decision rules.

The design of the ranked layers from the logical elements  $E_i$  (6) allows the number of the estimated parameters to be decreased in comparison with the formal neurons  $FN_i$  (5). Also, the computation of the ranked rules is easier for the elements  $E_i$  (6). On the other hand, this diminishes the discrimination power of the decision rules and it may be impossible to design the fully separable, ranked layer for some data sets by using only the elements  $E_i$  (6). This fact could be observed even for the Iris data sets.

## References

1. U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, *Advances in Knowledge Discovery and Data Mining*, AAAI Press/ The MIT Press 1996
2. O. R. Duda and P. E. Hart: *Pattern Classification and Scene Analysis*. New York, J. Wiley 1973
3. Ch. M. Bishop, *Neural networks for pattern recognition*, Clarendon Press, Oxford 1995
4. B. D. Ripley *Pattern Recognition and Neural Networks*, Cambridge University Press, 1996
5. L. Bobrowski, "The Ranked Networks of Formal Neurons". *Biocybernetics and Biomedical Engineering*, Vol. 12, Nr. 1 - 4, pp.61-75, 1992
6. J. Fuhrkranz: *Separate-and-Conquer Rule Learning*, Austrian Research Institute for Artificial Intelligence, Technical Report OEF AJ-TR-96-25  
<http://www.research.att.com/~wcohen/cs536/oefai-tr-96-25.ps>
7. M. Mizard and J.-P. Nadal: *Learning in Feedforward Layered Neural Networks: The Tiling Algorithm*, *J. Phys. A*22, 2191 (1889)
8. L. Bobrowski, H. Wasyluk, "Neural networks in discrimination of liver diseases", *Biocybernetics and Biomedical Engineering*, 1998 (in a press)
9. L. Bobrowski (Ed.): "Hepar - computer system for diagnosis support and data analysis" (in Polish), *Prace IBIB*, 31, 1992
10. H. Wasyluk, "The Four Year's Experience with HEPAR - Computer Assisted Diagnostic Program" in *MEDINFO 95 Proceedings*, R. A. Greenes et al. (Eds.)
11. L. Bobrowski, "Piecewise-Linear Classifiers, Formal Neurons and separability of the Learning Sets", *Proceedings of ICPR'96*, pp. 224-228 (*13th International Conference on Pattern Recognition*), August 25-29, 1996, Vienna, Austria)
12. L. Bobrowski: "Design of piecewise linear classifiers from formal neurons by some basis exchange technique". *Pattern Recognition*, 24(9), pp. 863-870, 1991.