Induction of Linear Separability Through the Ranked Layers of Binary Classifiers

Leon Bobrowski

Faculty of Computer Science, Bialystok Technical University,
ul. Wiejska 45A, Bialystok
and
Institute of Biocybernetics and Biomedical Engineering, PAS, Warsaw, Poland
e-mail: leon@ibib.waw.pl

Abstract: The concept of *linear separability* is used in the theory of neural networks and pattern recognition methods. This term can be related to examination of learning sets (classes) separation by hyperplanes in a given feature space. The family of *K* disjoined learning sets can be transformed into K linearly separable sets by the ranked layer of binary classifiers. Problems of the ranked layers deigning are analyzed in the paper.

Keywords. Learning sets, linear separability, formal neurons, binary classifiers, ranked

1 Introduction

The *Perceptron* model and the error-correction learning algorithm of formal neurons played a fundamental role in the early neural networks [1], [2]. The key concept here was the linear separability of learning sets. The convergence in a finite number of steps of the classical error correction algorithm depends on the linear separability of learning sets. Introducing a positive margin into error correction procedure allowed to ensure the convergence in a finite number of steps of the modified learning algorithms, also when the learning sets are not linearly separable [3], [4]. The perceptron criterion function linked to the error error-correction learning algorithm belongs to the family of *convex and piecewise linear* (*CPL*) criterion functions. Minimizing the *CPL* functions allows for, inter alia, effective designing of linear classifiers, linear forecasting models designing or carrying out selection of features subsets by using the *relaxed linear separability* (*RLS*) method [5].

The most popular algorithms currently used in data mining are the *support vector machines* (*SVM*) [6]. The *SVM* algorithms are also linked to linear separability of learning sets. An essential part of these algorithms is the linear separability induction through the application of kernel functions. The selection of the appropriate kernel functions is still an open and difficult problem in many practical problems.

A family of K disjoined learning sets can be transformed into K linearly separable sets as a result of the transformation by ranked layer of formal neurons [4], [7]. This result can be extended on the ranked layer of arbitrary binary classifiers. The proof of the theorem concerning induction of linear separability through ranked layer of binary classifiers is given in this paper.

2 Separable Learning Sets

Let us assume that m objects O_j ($j = 1,\ldots,m$) are represented as the so called feature vectors $\mathbf{x}_j = [x_{j1},\ldots,x_{jn}]^T$, or as points in the n-dimensional feature space $F[n]$ ($\mathbf{x}_j \in F[n]$). Components (*features*) x_i of the feature vector $\mathbf{x}$ represent numerical results of different measurements on a given object O ($x_i \in \{0,1\}$ or $x_i \in R$).

We assume that the feature vector $\mathbf{x}_j(k)$ ($j = 1,\ldots\ldots, m$) has been labelled in accordance with the object $O_j(k)$ *category* (*class*) ω_k ($k = 1,\ldots,K$). The learning set C_k contains m_k feature vectors $\mathbf{x}_j(k)$ assigned to the k-th category ω_k

$$C_k = \{\mathbf{x}_j(k)\} \quad (j \in I_k) \tag{1}$$

where I_k is the set of indices j of the feature vectors $\mathbf{x}_j(k)$ assigned to the class ω_k.

Definition 1. The learning sets C_k (1) are *separable* in the feature space $F[n]$, if they are disjoined in this space ($C_k \cap C_{k'} = \varnothing$, if $k \neq k'$). This means that the feature vectors $\mathbf{x}_j(k)$ and $\mathbf{x}_{j'}(k')$ belonging to different learning sets C_k and $C_{k'}$ cannot be equal:

$$(k \neq k') \Rightarrow (\forall j \in I_k) \; and \; (\forall j' \in I_{k'}) \;\; \mathbf{x}_j(k) \neq \mathbf{x}_{j'}(k') \tag{2}$$

We are also considering separation of the sets C_k (1) by the hyperplanes $H(\mathbf{w}_k,\theta_k)$ in the feature space $F[n]$:

$$H(\mathbf{w}_k,\theta_k) = \{\mathbf{x}: \mathbf{w}_k^T\mathbf{x} = \theta_k\}. \tag{3}$$

where $\mathbf{w}_k = [w_{k1},\ldots,w_{kn}]^T \in R^n$ is the weight vector, $\theta_k \in R^1$ is the threshold, and $(\mathbf{w}_k)^T\mathbf{x}$ is the inner product.

Definition 2. The feature vector $\mathbf{x}_j$ is situated on the *positive side* of the hyperplane $H(\mathbf{w}_k,\theta_k)$ (3) if and only if $(\mathbf{w}_k)^T\mathbf{x}_j > \theta_k$. Similarly, vector $\mathbf{x}_j$ is situated on the *negative side* of $H(\mathbf{w}_k,\theta_k)$ if and only if $(\mathbf{w}_k)^T\mathbf{x}_j < \theta_k$.

Definition 3. The learning sets (1) are *linearly separable* in the n-dimensional feature space $F[n]$ if each of the sets C_k can be fully separated from the sum of the remaining sets C_i by some hyperplane $H(\mathbf{w}_k,\theta_k)$ (3):

$$\begin{aligned}&(\exists k \in \{1,\ldots,K\}) \; (\exists\, \mathbf{w}_k,\theta_k) \; (\forall \mathbf{x}_j(k) \in C_k) \;\; \mathbf{w}_k^T\mathbf{x}_j(k) > \theta_k. \\ &\qquad\qquad \textbf{\textit{and}} \;\; (\forall \mathbf{x}_j(i) \in C_i,\, i \neq k) \;\; \mathbf{w}_k^T\mathbf{x}_j(i) < \theta_k\end{aligned} \tag{4}$$

In accordance with the inequalities (4), all the vectors $\mathbf{x}_j(k)$ from the set C_k are situated on the positive side of the hyperplane $H(\mathbf{w}_k,\theta_k)$ (3) and all the vectors $\mathbf{x}_j(i)$ from the remaining sets C_i are situated on the negative side of this hyperplane.

3 Layers of Binary Classifiers

The binary classifiers $BC_i(\mathbf{v})$ operate on feature vectors $\mathbf{x}$ ($\mathbf{x} \in F[n]$) and are characterized by such a decision rule $r_i(\mathbf{v};\ \mathbf{x})$ which depends on the vector of parameters $\mathbf{v}$ ($\mathbf{v} \in R^N$) and has the binary output of $r_i = r_i\ (\mathbf{v};\ \mathbf{x})$ ($r_i \in \{0, 1\}$, $i = 1,\ldots,m$).

Definition 4. The *activation field* $A_i(\mathbf{v})$ of the i-th binary classifier $BC_i(\mathbf{v})$ with the decision rule $r_i(\mathbf{v};\mathbf{x})$ is the set of such a feature vectors $\mathbf{x}$ which activate this classifier

$$A_i(\mathbf{v}) = \{\mathbf{x}\colon r_i(\mathbf{v};\ \mathbf{x}) = 1\} \tag{5}$$

A layer composed of L binary classifiers $BC_i(\mathbf{v}_i)$ with the decision rules $r_i(\mathbf{v}_i;\mathbf{x})$ produces output vectors $\mathbf{r}(\mathbf{V};\mathbf{x}) = [r_1(\mathbf{v}_1;\mathbf{x}),\ldots,r_L(\mathbf{v}_L;\mathbf{x})]^T$, where $\mathbf{V} = [\mathbf{v}_1^T,\ldots,\mathbf{v}_{L'}^T]^T$. The layer of L binary classifiers $BC_i(\mathbf{v}_i)$ transforms feature vectors $\mathbf{x}_j(k)$ from the learning sets C_k (1) into the sets R_k of the binary output vectors $\mathbf{r}_j(k) = \mathbf{r}(\mathbf{V};\mathbf{x}_j(k))$.

$$R_k = \{\mathbf{r}_j(k)\} \quad (j \in I_k) \tag{6}$$

Under certain conditions, it is possible to achieved the linear separability (4) of the sets R_k (6). These conditions are analyzed in the paper.

Consider a few examples of binary classifiers $BC_i(\mathbf{v}_i)$.

i. Formal neurons $FN(\mathbf{w}_i, \theta)$

The formal neuron $FN(\mathbf{w}, \theta)$ with the weight vector $\mathbf{w} = [w_1,\ldots,w_n]^T \in R^n$ and the threshold θ ($\theta \in R$) can be characterized by the below decision rule $r_{FN}(\mathbf{w}, \theta;\ \mathbf{x})$:

$$\textbf{\textit{if}}\ \mathbf{w}^T\mathbf{x} > \theta\ \textbf{\textit{then}}\ r_{FN}(\mathbf{w}, \theta;\ \mathbf{x}) = 1,\ \textbf{\textit{else}}\ r_{FN}(\mathbf{w}, \theta;\ \mathbf{x}) = 0 \tag{7}$$

The formal neuron $FN(\mathbf{w},\theta)$ is activated ($r_{FN}(\mathbf{w}, \theta;\ \mathbf{x}) = 1$) if and only if the weighed sum $w_1x_1+\ldots+ w_n x_n$ of n inputs x_i ($x_i \in R$) is greater than the threshold θ. This decision rule $r_{FN}(\mathbf{w}, \theta;\ \mathbf{x})$ depends on $n + 1$ parameters w_i ($i = 1,\ldots,n$) and θ. The activation field $A_i(\mathbf{v})$ (5) is in this case the half space $A_{FN}(\mathbf{w},\theta) = \{\mathbf{x}\colon \mathbf{w}^T\mathbf{x} > \theta\}$.

ii. Logical elements $LE_i(\mathbf{w}, \theta)$

The i-th *logical element* $LE_i(\mathbf{w},\theta)$ ($i = 1,\ldots, L$) can be defined as the formal neuron (7) with only one entry $x_{k(i)}$ ($x_{k(i)} \in R$), where $k(i)$ is the number of the distinguished entry $k(i) \in \{1,\ldots,n\}$). The decision rule $r_{LEi}(w_{k(i)},\theta_i;\ \mathbf{x})$ of the i-th logical element $LE_i(\mathbf{w},\theta)$ can be given similarly to (7) as:

$$\textbf{\textit{if}}\ w_{k(i)}x_{k(i)} > \theta\ \textbf{\textit{then}}\ r_{LEi}(w_{k(i)},\theta_i;\ \mathbf{x}) = 1,\ \textbf{\textit{else}}\ r_{LEi}(w_{k(i)},\theta_i;\ \mathbf{x}) = 0 \tag{8}$$

The above decision rule $r_{LEi}(\mathrm{w}_{k(i)},\theta_i;\ \mathbf{x})$ depends on two parameters: $\mathrm{w}_{k(i)}$ and θ_i. The hyperplane $H(\mathrm{w}_{k(i)},\theta_i) = \{\mathbf{x}\colon \mathrm{w}_{k(i)}\mathrm{x}_{k(i)} = \theta_i\}$ (3) in the feature space $F[n]$ is parallel to all axes x_l except the axis $x_{k(i)}$ ($l = 1,...,n$ *and* $l \neq k(i)$) and perpendicular to the axis $x_{k(i)}$.

iii. Radial binary classifiers $RC(\mathbf{w}_{0i},\rho_i)$

The decision rule $r_{RC}(\mathbf{w}_0,\rho;\ \mathbf{x})$ of the radial classifier $RC(\mathbf{w}_0,\rho)$ can be defined in the below manner by using the distance $\delta(\mathbf{w}_0,\mathbf{x})$ between the point $\mathbf{x}$ and the center of the ball $\mathbf{w}_0 = [\mathrm{w}_{01},\ldots,\mathrm{w}_{0n}]$.

$$\textbf{\textit{if}}\ \ \delta(\mathbf{w}_0,\mathbf{x}) \leq \rho\ \ \textbf{\textit{then}}\ r_{RC}(\mathbf{w}_0,\rho;\ \mathbf{x}) = 1,\ \textbf{\textit{else}}\ r_{RC}(\mathbf{w}_0,\rho;\ \mathbf{x}) = 0 \tag{9}$$

The radial classifier $RC(\mathbf{w}_0,\rho)$ is excited ($r_{RC}(\mathbf{w}_0,\rho;\ \mathbf{x}) = 1$) if and only if the distance $\delta(\mathbf{w}_0,\mathbf{x})$ is no greater than the radius ρ. The decision rule $r_{RC}(\mathbf{w}_0,\rho;\ \mathbf{x})$ (9) of the radial classifier $RC(\mathbf{w}_0,\rho)$ depends on the $n + 1$ parameters $\mathbf{w}_0 = [\mathrm{w}_{01},\ldots,\mathrm{w}_{0n}]^T$ and ρ. The activation field $A_{RC}(\mathbf{w}_0,\ \rho)$ (5) of the radial classifier $RC(\mathbf{w}_0,\rho)$ is the ball with the center $\mathbf{w}_0$ and the radius ρ. The shape of this activation field $A_{RC}(\mathbf{w}_0,\ \rho)$ depends of the choice of the distance function $\delta(\mathbf{w}_0,\mathbf{x})$. A few examples of distance functions $\delta(\mathbf{w}_0,\mathbf{x})$ are given below [3]:

$$\begin{aligned}
&\delta_E(\mathbf{w}_0,\mathbf{x}) = ((\mathbf{x} - \mathbf{w}_0)^T(\mathbf{x} - \mathbf{w}_0))^{1/2} && \text{- the } \textit{Euclidean} \text{ distance}\\
&\delta_M(\mathbf{w}_0,\mathbf{x}) = ((\mathbf{x} - \mathbf{w}_0)^T\boldsymbol{\Sigma}^{-1}(\mathbf{x} - \mathbf{w}_0))^{1/2} && \text{- the } \textit{Mahalanobis} \text{ distance}\\
&\delta_{L1}(\mathbf{w}_0,\mathbf{x}) = \sum_{i = 1,\ldots,N} |\mathrm{w}_{0i} - \mathrm{x}_i| && \text{- the } L_1 \text{ distance}
\end{aligned} \tag{10}$$

4 Properties of the Ranked Layers of Binary Classifiers

The strategy of the ranked layers designing allows to find a layer with such a property that the sets R_k (6) of transformed vectors $\mathbf{r}_j(k) = \mathbf{r}(\mathbf{V};\mathbf{x}_j(k))$ become linearly separable (4). The proposed multistage designing procedure involves finding a sequence of *admissible cuts* by binary classifiers $BC_i(\mathbf{v}_i)$ with the decision rules $r_i(\mathbf{v}_i;\ \mathbf{x})$ (5).

Definition 5. The binary classifier $BC_i(\mathbf{v}_i)$ with the decision rule $r_i(\mathbf{v}_i;\ \mathbf{x})$ (5) is *admissible* in respect to a given learning set $C_{k'}$ from the family (1) if and only if $n_{k'}$ ($n_{k'} > 0$) elements $\mathbf{x}_j(k')$ of this set activate the classifier and none of elements $\mathbf{x}_j(k)$ from other sets C_k ($k \neq k'$) activate the classifier $BC_i(\mathbf{v}_i)$

$$\begin{gathered}
(\forall k' \in \{1,\ldots,K\})\ (\exists \mathbf{x}_j(k') \in C_{k'})\ r_i(\mathbf{v}_i;\ \mathbf{x}_j(k')) = 1,\ \textit{and}\\
(\forall \mathbf{x}_j(k)\ \in C_k,\ \textit{where}\ k \neq k')\ r_i(\mathbf{v}_i;\ \mathbf{x}_j(k)) = 0
\end{gathered} \tag{11}$$

The activation field $A_i(\mathbf{v}_i)$ (4.1) of the admissible binary classifier $BC_i(\mathbf{v}_i)$ contains some elements $\mathbf{x}_j(k')$ of only one learning set $C_{k'}$ (1).

Let us describe the multistage procedure of the ranked layer designing from binary classifiers $BC_i(\mathbf{v})$ on the basis of the learning sets C_k (1).

Ranked layer designing procedure. (12)

Stage i. (Start)

- put $n = 1$ and define the sets $C_k[n]$: $(\exists k \in \{1,...,K\})$ $C_k[n] = C_k$

Stage ii. (Admissible classifier)

- find parameters $\mathbf{v}_{i(n)}$ of classifier $BC_{i(n)}(\mathbf{v}_{i(n)})$ admissible (11) to some set $C_{k(n)}[n]$.

Stage iii. (Admissible reduction (*cut*) of the set $C_{k(n)}[n]$)

- delay such feature vectors $\mathbf{x}_j$ which activate the classifier $BC_{i(n)}(\mathbf{v}_{i(n)})$

$$C_{k(n)}[n+1] = C_{k(n)}[n] - \{\mathbf{x}_j(k(n)) \in C_{k(n)}[n]: r_{i(n)}(\mathbf{v}_{i(n)}; \mathbf{x}_j(k(n)) = 1\} \quad (13)$$

Stage iv. (Stop criterion)

if all the sets $C_k[n+1]$ are empty ***then*** *stop*

else *increase* the index n by one ($n \rightarrow n+1$) ***and*** *go to* the *Stage ii.* □

It can be seen that each binary classifiers $BC_i(\mathbf{v}_i)$ added to the layer reduces (13) the set $C_{k(n)}[n]$ by at least one feature vector $\mathbf{x}_j(k(n))$. Based on this property, it can be proved that if the learning sets C_k (1) are separable (2), then after finite number L of steps the procedure will be stopped. The following Lemma results.

Lemma 2. The number L of binary classifiers $BC_i(\mathbf{v}_i)$ in the ranked layer is no less than the number K of the learning sets C_k (1) and no greater than the number m of the feature vectors $\mathbf{x}_j(k)$ in these sets.

$$K \leq L \leq m \quad (14)$$

The minimal number $L = K$ of binary classifiers $BC_i(\mathbf{v}_i)$ appears in the ranked layer when whole learning sets C_k (1) are reduced (13) during successive steps n. The maximal number $L = m$ of elements appears in the ranked layer when only single elements $\mathbf{x}_j(k)$ are reduced during successive steps n.

In accordance with the postulate of *a large admissible reduction,* the number $n_{k(n)}[n]$ of the reduced elements $\mathbf{x}_j(k(n))$ ($x_j(k(n) \in C_{k(n)}[n]$) during each step n (13) should be as large as possible [4]. In order to fulfill this postulate the below rule can be introduced:

The set $C_{k(n)}[n]$ reduced during the n-th designing step should be characterized by the largest number $n_{k(n)}[n]$ of reduced elements $\mathbf{x}_j(k(n))$: (15)

$$(\forall k \in \{1,\ldots,K\}) \;\; n_{k(n)}[n] \geq n_k[n]$$

The sequence of the parameters $\mathbf{v}_{i(1)}$ appears as the result of the procedure (12):

$$\mathbf{v}_{i(1)}, \mathbf{v}_{i(2)}, \ldots, \mathbf{v}_{i(L)} \quad (16)$$

Additionally, the class indices $k(n)$ of the sets $C_{k(n)}[n]$ reduced (13) during successive steps n can be obtained

$$k(1), k(2),..., k(L) \tag{17}$$

The parameters $\mathbf{v}_{i(n)}$ (16) define the decision rule $r_{i(n)} = r_{i(n)}(\mathbf{v}_{i(n)};\mathbf{x})$ ($r_{i(n)} \in \{0, 1\}$) of the binary classifiers $BC_{i(n)}(\mathbf{v}_{i(n)})$ in the ranked layer

$$BC_{i(1)}(\mathbf{v}_{k(1)}), BC_{i(2)}(\mathbf{v}_{i(2)}),..., BC_{i(L)}(\mathbf{v}_{i(L)}) \tag{18}$$

Feature vectors $\mathbf{x}_j(k)$ from the learning sets C_k (1) are transformed by the ranked layer of L classifiers (18) into vectors $\mathbf{r}_j(k) = [r_{j1},\ldots,r_{jL}]^T$ with L binary components r_{ji} ($r_{ji} \in \{0, 1\}$) which are related to the category ω_k.

$$\begin{aligned}&(\forall k\in \{1,\ldots,K\})\ (\exists \mathbf{x}_j(k) \in C_k)\\ &\qquad \mathbf{r}_j(k) = [r_{i(1)}(\mathbf{v}_{i(1)};\mathbf{x}_j(k)),\ldots, r_{i(L)}(\mathbf{v}_{i(L)};\mathbf{x}_j(k))]^T\end{aligned} \tag{19}$$

The transformed vectors $\mathbf{r}_i(k)$ (19) can be represented in the below manner by L – dimensional vectors $\mathbf{q}_n = [q_{n1},\ldots, q_{nL}]^T$ ($l = 1,\ldots, L$) with the binary components q_{lj}

$$\begin{aligned}&\mathbf{q}_1 = [1, q_{12},\ldots,r_{1L}]^T\\ &\mathbf{q}_2 = [0,1, q_{23},\ldots,r_{2L}]^T\\ &\ldots\ldots.\\ &\ldots\ldots.\\ &\mathbf{q}_L = [0,0,\ldots,0,1]^T\end{aligned} \tag{20}$$

It is assumed in the above representation that each component q_{nj} with $j > n$ can be set to any binary value $q_{nj} = 1$ or $q_{nj} = 0$. In result, each vector $\mathbf{q}_n$ (20) can represent more than one transformed vector $\mathbf{r}_j(k)$ (19).

The vectors $\mathbf{q}_n = [q_{n,1},\ldots,q_{n,L}]^T$ (20) can be used in the below decision rule, which results from the procedure (12) of the ranked layers designing

$$\textbf{\textit{if}}\ \ (\forall i\text{: } 0 < i < n)\ \ q_{n,i} = 0\ \ \textbf{\textit{and}}\ q_{n,n} = 1,\ \textbf{\textit{then}}\ \ \mathbf{q}_n \in \omega_{k(n)} \tag{21}$$

where $k(n)$ is the index (17) of the set $C_{k(n)}$ reduced during the n–th stage.

The rule (21) links the vectors $\mathbf{q}_n$ (20) to particular classes ω_k. This rule is based only on the content of the n-th admissible cut (13) of the set $C_{k(n)}$. In this manner, each class ω_k can be represented by the set Q_k of the related vectors $\mathbf{q}_n(k)$ (20).

$$Q_k = \{\mathbf{q}_n(k)\text{: } k(n) = k\}, \tag{22}$$

where $k \in \{1,\ldots, K\}$ and the distinguished index $k(n)$ is determined by (17).

Theorem 1. The sets Q_k (22) are linearly separable (4).

Proof: The proof is based on the example of such parameters $\mathbf{w}_k$, θ_k, which fulfil the inequalities (4) [7]. Let us introduce the L-dimensional vector $\mathbf{a} = [a_1,\ldots,a_L]^T$ with the components a_i specified below

$$(\forall i \in \{1,\ldots, L\}) \quad a_i = 1/2^i \tag{23}$$

The weight vector $\mathbf{w}_k = [w_{k1},\ldots,w_{kL}]^T$ is defined on the basis of the sequence (17) in accordance with the below rule

$$(\forall i \in \{1,\ldots, L\}) \ \textbf{\textit{if}} \ k(i) = k, \textbf{\textit{then}} \ w_{ki} = a_i \ \textbf{\textit{else}} \ w_{ki} = - a_i \tag{24}$$

It can be directly verified that all the inequalities (4) are fulfilled by the vectors $\mathbf{w}_k$ with the components w_{ki} (24) and the threshold $\theta_k = 0$. This means that the sets Q_k (22) are linearly separable (4). □

Corollary. The sets $R_k = \{\mathbf{r}_j(k)\}$ (6) of the transformed vectors $\mathbf{r}_j(k)$) related to particular categories ω_k are linearly separable (4).

$$\begin{aligned} &(\forall k \in \{1,\ldots, K\}) \ (\exists \mathbf{w}_k) \ (\forall \mathbf{r}_j(k) \in R_k) \quad (\mathbf{w}_k)^T\mathbf{r}_j(k) > 0 \\ &\qquad \textbf{\textit{and}} \quad (\forall \mathbf{r}_j(k') \in R_{k'}, k \neq k') \quad (\mathbf{w}_k)^T\mathbf{r}_j(k') < 0 \end{aligned} \tag{25}$$

The transformation (25) of the feature vectors $\mathbf{x}_j(k)$ by the ranked layer allows to replace the separable (2) learning sets C_k (1) by the sets $R_k = \{\mathbf{r}_j(k)\}$ (6) which are linearly separable (25). Let us remark that the *Theorem* 1 is also valid if the ranked layer is composed of different types of binary classifiers $BC_i(\mathbf{v}_i)$.

5 Examples of Implementations of the Ranked Layer Designing Procedure

Let us consider at the beginning, the procedure ranked layer designing from logical elements $LE_i(\mathbf{w},\theta)$ (8). The decision rule $r_{LEi}(w_{k(i)},\theta_i;\mathbf{x})$ (8) of the i-th logical element $LE_i(\mathbf{w},\theta)$ depends on only one feature $x_{k(i)}$. The decision rule $r_{LEi}(w_{k(i)},\theta_i;\mathbf{x})$ (8) can be decomposed into two rules $r_i^+(\mathbf{x})$ and $r_i^-(\mathbf{x})$ and represented in the below manner for each feature x_i ($i = 1,\ldots,n$):

$$(\forall i \in \{1,\ldots,n\}) \ \textbf{\textit{if}} \ x_i > \theta_i^+, \ \textbf{\textit{then}} \ r_i^+(\mathbf{x}) = 1 \ \textbf{\textit{else}} \ r_i^+(\mathbf{x}) = 0 \tag{26}$$

and

$$(\forall i \in \{1,\ldots,n\}) \ \textbf{\textit{if}} \ x_i < \theta_i^-, \ \textbf{\textit{then}} \ r_i^-(\mathbf{x}) = 1 \ \textbf{\textit{else}} \ r_i^-(\mathbf{x}) = 0 \tag{27}$$

where θ_i^+ and θ_i^- are two numerical parameters ($\theta_i^+ \in R^1$ and $\theta_i^- \in R^1$).

In accordance with the *admissible reduction* principle (11), the parameters θ_i^+ and θ_i^- should ensure the homogenous reduction of the maximal numbers $n^+(i)$ and $n^-(i)$ of elements $\mathbf{x}_j(k)$ belonging to only one learning set C_k (1). The term *homogenous* *reduction* (11) means that the condition $r_i^+(\mathbf{x}) = 1$ (26) is fulfilled by $n^+(i)$ elements $\mathbf{x}_j(k)$ of only one learning set C_k (1). Similarly, the condition $r_i^-(\mathbf{x}) = 1$ (27) should be fulfilled by $n^-(i)$ elements $\mathbf{x}_j(k')$ of only one learning set $C_{k'}$ (1).

In order to satisfy the postulate of *a large admissible reduction* the below rules can be used

$$(\forall i \in \{1,\ldots,n\})\ \ n(i) = max\{\, n^+(i),\, n^-(i)\},\ and \qquad n(i') = max\{\, n(1),\ldots,\, n(L)\}, \tag{28}$$

where $n(i)$ is the maximal number of elements $\mathbf{x}_j(k)$ from one learning set C_k (1), which can be reduced by using the i-th feature x_i (coordinates x_i)and the rule (26) or (27), i' is the index of such a coordinate $x_{i'}$ which allows to reduce the maximal number $n(i')$ of elements $\mathbf{x}_j(k')$ belonging to only one learning set $C_{k'}$ (1).

The rule (28) allows to determine the class indices $k(n)$ (17) of the sets $C_{k(n)}[n]$ reduced (13) during successive steps n of the *Ranked layer designing procedure* (12). The optimal threshold value $\theta(n)$ can be determined on the basis of the rules (26) and (27). The optimal threshold value $\theta(n)$ should result in the maximal number $n(i')$ (28) of elements $\mathbf{x}_j(k)$ of one learning set C_k (1) reduced during each step n. The rules (28) and include the selection of the optimal threshold values θ_i^+ and θ_i^- for each coordinate x_i. The optimal threshold values θ_i^+ (26) and θ_i^-(27) should result in the maximal numbers $n^+(i)$ and $n^-(i)$ of the reduced elements $\mathbf{x}_j(k)$ of one learning set C_k (1). The implementation of the ranked layers design rules is relatively simple in the case of logical elements $LE_i(\mathbf{w},\theta)$ (8). For small data sets (1), such rules can even be implemented without a computer, based on handwritten calculations.

Let us consider now the layer of formal neurons $FN(\mathbf{w}_i, \theta_i)$ (7) which can be linked to the hyperplanes $H(\mathbf{w}_i,\theta_i)$ (3).

Definition 6. The hyperplane $H(\mathbf{w}_{k'},\theta_{k'})$ (3) is *admissible* (*Definition* 5) to a given set $C_{k'}[n]$ from the family $\{C_k[n]\}$ (13) if and only if $n_{k'}$ ($n_{k'} > 0$) elements $\mathbf{x}_j(k')$ of this set is situated on the positive side of the hyperplane $H(\mathbf{w}_{k'},\theta_{k'})$ and all elements $\mathbf{x}_j(k)$ from other sets C_k ($k \neq k'$) are situated on the negative side of this hyperplane:

$$(\forall k' \in \{1,\ldots,K\})\ (\exists \mathbf{x}_j(k') \in C_{k'}[n])\ \ \mathbf{w}_{k'}^T\mathbf{x}_j(k') > \theta_{k'},\ and \qquad (\forall \mathbf{x}_j(k) \in C_k[n],\ where\ k \neq k')\ \ \mathbf{w}_{k'}^T\mathbf{x}_j(k) < \theta_{k'} \tag{29}$$

The sequence of admissible hyperplanes $H(\mathbf{w}_i,\theta_i)$ (3) should be consisted with the *Ranked layer designing procedure* (12). The parameters $\mathbf{w}_i$ and θ_i of admissible hyperplanes $H(\mathbf{w}_i,\theta_i)$ (3) define the formal neurons $FN(\mathbf{w}_i,\theta_i)$ (7) of the ranked layer.

The hyperplane $H(\mathbf{w}_{k'},\theta_{k'})$ (3) admissible (*Definition* 5) to a given set $C_{k'}[n]$ from the family $\{C_k[n]\}$ (13) can be obtained through minimization of the convex and piecewise linear (*CPL*) criterion functions $\Psi_{k''}(\mathbf{w},\theta)$ [7]. The preceptron criterion function belongs to the *CPL* family [4]. The criterion function $\Psi_{k'}(\mathbf{w},\theta)$ can be defined on the basis of the positive G_k^+ and the negative G_k^- sets of feature vectors $\mathbf{x}_j$ from the sets $\{C_k[n]\}$ (13)

$$G_{k'}^+ = \{\mathbf{x}_j\colon \mathbf{x}_j \in C_{k'}[n]\} \;\; \textit{and} \;\; G_{k'}^- = \{\mathbf{x}_j\colon \mathbf{x}_j \in \bigcup_{k \neq k'} C_k[n]\} \tag{30}$$

Each element $\mathbf{x}_j$ of the set $G_{k'}^+$ defines the positive penalty function $\varphi_j^+(\mathbf{w},\theta)$

$$\begin{aligned} &(\forall \mathbf{x}_j \in G_{k'}^+) \\ &\quad \textbf{\textit{if}} \;\; \mathbf{w}^T\mathbf{x}_j - \theta \leq 1, \textbf{\textit{then}} \;\; \varphi_j^+(\mathbf{w},\theta) = 1 - \mathbf{w}^T\mathbf{x}_j + \theta, \textbf{\textit{else}} \;\; \varphi_j^+(\mathbf{w},\theta) = 0 \end{aligned} \tag{31}$$

Similarly, each element $\mathbf{x}_j$ of the set $G_{k'}^-$ defines the negative penalty function $\varphi_j^-(\mathbf{w},\theta)$

$$\begin{aligned} &(\forall \mathbf{x}_j \in G_{k'}^-) \\ &\quad \textbf{\textit{if}} \;\; \mathbf{w}^T\mathbf{x}_j - \theta \geq -1, \textbf{\textit{then}} \;\; \varphi_j^-(\mathbf{w},\theta) = \mathbf{w}^T\mathbf{x}_j - \theta \;\; \geq -1, \textbf{\textit{else}} \;\; \varphi_j^-(\mathbf{w},\theta) = 0 \end{aligned} \tag{32}$$

The criterion functions $\Psi_{k'}(\mathbf{w},\theta)$ is the sum of the functions $\varphi_j^+(\mathbf{w},\theta)$ and $\varphi_j^-(\mathbf{w},\theta)$

$$\Psi_{k'}(\mathbf{w},\theta) = \Sigma \, \varphi_j^+(\mathbf{w},\theta) + \lambda \, \Sigma \, \varphi_j^-(\mathbf{w},\theta) \tag{33}$$

where λ ($\lambda > 0$) is a positive parameter (*price*).

Minimization of the function $\Psi_{k'}(\mathbf{w},\theta)$ allows to find optimal parameters $(\mathbf{w}_{k'}^*,\theta_{k'}^*)$:

$$\min \Psi_{k'}(\mathbf{w},\theta) = \Psi_{k'}(\mathbf{w}_{k'}^*,\theta_{k'}^*) \geq 0 \tag{34}$$

The basis exchange algorithms which are similar to the linear programming allow to find efficiently minimum $\Psi_{k'}(\mathbf{w}_{k'}^*,\theta_{k'}^*)$ of the criterion function $\Psi_{k'}(\mathbf{w},\theta)$ (33) [7]. It can be shown that the optimal parameters $(\mathbf{w}_{k'}^*,\theta_{k'}^*)$ (34) obtained from the function $\Psi_{k'}(\mathbf{w},\theta)$ (39) with a sufficiently large parameter λ define the admissible hyperplane $H(\mathbf{w}_{k'}^*,\theta_{k'}^*)$ (29) [4]. The optimal parameters $(\mathbf{w}_{k'}^*,\theta_{k'}^*)$ allows to define the formal neurons $FN(\mathbf{w}_{k'}^*,\theta_{k'}^*)$ (7) of the ranked layer in accordance with the *Ranked layer designing procedure* (12).

6 Concluding Remarks

Separable learning sets C_k (2) can always be transformed into such sets R_k (6) which are linearly separable (4). The linear separability is induced through the transformation of elements $\mathbf{x}_j(k)$ of the learning sets C_k (1) by the ranked layer of binary classifiers $BC_i(\mathbf{v}_i)$. The paper contains a proof of this property.

The paper also contains examples of designing ranked layers of formal neurons *FN*($\mathbf{w}_i$, θ_i) (7) and logical elements *LE*$_i$($\mathbf{w}_i$,θ_i) (8). Efficient designing ranked layers of radial classifiers *RC*($\mathbf{w}_0$,ρ) (9) is still an open problem from computational point of view.

Acknowledgment. This work was supported by the by the NCBiR project N R13 0014 04, and partially financed by the project S/WI/2/2011 from the Białystok University of Technology, and by the project 16/St/2011 from the Institute of Biocybernetics and Biomedical Engineering PAS.

References

1. Rosenblatt F.: Principles of neurodynamics, Spartan Books, Washington (1962)
2. Minsky M. L. and Papert S. A.: Perceptrons, MIT Press, Cambridge, MA 1969
3. Duda, O. R., Hart, P. E., and Stork, D. G.: Pattern classification, J. Wiley, New York, (2001)
4. Bobrowski L.: Eksploracja danych oparta na wypukłych i odcinkowo-liniowych funkcjach kryterialnych (Data mining based on convex and piecewise linear (*CPL*) criterion functions) (*in Polish*), Technical University Białystok (2005)
5. Bobrowski, L., Łukaszuk, T: "Feature selection based on relaxed linear separability", Biocybernetics and Biomedcal Engineering, Volume 29, Number 2, pp. 43-59 (2009)
6. Vapnik V. N.: Statistical Learning Theory, J. Wiley, New York, 1998
7. Bobrowski L.: "Design of piecewise linear classifiers from formal neurons by some basis exchange technique", Pattern Recognition, **24**(9), pp. 863-870 (1991).