Kybernetika

Jiří Grim

Neuromorphic features of probabilistic neural networks

Kybernetika, Vol. 43 (2007), No. 5, 697--712

Persistent URL: `http://dml.cz/dmlcz/135807`

KYBERNETIKA — VOLUME *43* (2007), NUMBER 5, PAGES 697 – 712

NEUROMORPHIC FEATURES OF PROBABILISTIC NEURAL NETWORKS

Jiří Grim

We summarize the main results on probabilistic neural networks recently published in a series of papers. Considering the framework of statistical pattern recognition we assume approximation of class-conditional distributions by finite mixtures of product components. The probabilistic neurons correspond to mixture components and can be interpreted in neurophysiological terms. In this way we can find possible theoretical background of the functional properties of neurons. For example, the general formula for synaptical weights provides a statistical justification of the well known Hebbian principle of learning. Similarly, the mean effect of lateral inhibition can be expressed by means of a formula proposed by Perez as a measure of dependence tightness of involved variables.

Keywords: probabilistic neural networks, distribution mixtures, sequential EM algorithm, pattern recognition

AMS Subject Classification: 62P10, 62H30, 68T10

1. INTRODUCTION

In the last decades a lot of knowledge has been accumulated about neural systems. Many details are known about the internal structure and interconnections of neurons and also about the adaptive processes which are responsible for learning of neurons. Different neural network models have been proposed to explain the functioning of neural assemblies and simultaneously they have been successfully applied to practical problems. At present artificial neural networks can be viewed as a standard tool to solve difficult problems in pattern recognition and in many other fields.

On the other hand even very successful neural network models seem to be poor in comparison with the excellent functioning of biological neural systems. As it appears, we are still missing the full understanding of the basic principles which guarantee the functional superiority of biological neural systems in spite of the fact that they consist of imprecise, noisy and unreliable biological neurons. Unlike different training algorithms, biological neural systems have only limited possibilities to control and coordinate the functioning of individual neurons centrally, according to some global criteria. In this sense the decentralized learning of artificial neural networks based on the modular adaptivity of neurons also represents one of the essential steps to be done towards better explanation of biological neural network principles.

The main motivation of the considered probabilistic approach to neural networks is to clarify basic functional properties of biological neural systems by formal interpretation of a general statistical classification method. The classification of complex stimuli into different categories undoubtedly represents an important activity permanently performed by biological neural systems. It is therefore of theoretical relevance to demonstrate that a reliable statistical classification method can be realized by means of highly autonomous adaptive neurons.

The probabilistic approach to neural networks has been developed in the framework of statistical pattern recognition. The term probabilistic neural network (PNN) has been suggested by Specht in connection with kernel-estimates of class-conditional densities [26]. In the following we refer mainly to our early paper [4] and other results published in the last years ([6] – [20]). Considering PNN we approximate the underlying probability distributions of data by finite mixtures which can be optimized by means of EM algorithm (cf. e. g. [2, 16, 22]). In this sense PNN do not provide a new technique of statistical pattern recognition but they may substantially contribute to better understanding of the functional principles of biological neural networks.

The main idea of PNN is to view the components of mixtures as neurons. Unlike other authors (cf. e. g. [21, 27, 29]) we assume the mixture components in product form which is essential from the point of view of biological interpretation. In the published papers we have verified different neuromorphic features of PNN and their compatibility with biological neural networks.

In particular we have shown that multilayer PNN can be designed sequentially by means of information preserving transform which is defined for each layer by the respective estimated mixture parameters [7, 8, 9, 28]. The principle of information preserving transform is closely related to data reduction problems studied by Perez and particularly to his concept of ε-sufficiency [23, 24]. The interconnection structure of multilayer PNN may be incomplete and can be optimized in a statistically correct way by using product mixtures with structural parameters [8, 16]. The structural mixture model is one of the few statistically correct subspace approaches to pattern recognition [16]. Independently trained PNN can be combined both horizontally and vertically [14, 15, 16] in a way which corresponds with the parallel and multilayer structures occurring in biological neural networks. Simultaneously, the PNN can be constructed in a modular way assuming autonomous properties of probabilistic neurons [12].

The probabilistic neuron can be interpreted in neurophysiological terms at the level of functional properties of a biological neuron [12, 13]. Thus e. g. the resulting theoretical model of synaptical plasticity justifies the well known Hebbian principle of learning. Weighting of data is compatible with probabilistic neural networks [18] in a way allowing for selective evaluation of training data (emotional learning). In case of infinite training data sequence the EM algorithm can be realized as a sequential procedure [10, 13] which can be viewed as one infinite iteration of EM algorithm with periodic updating of the estimated parameters.

2. MIXTURE BASED STATISTICAL PATTERN RECOGNITION

Considering the framework of statistical pattern recognition we assume that some multivariate observations have to be classified with respect to a finite set of classes $\Omega = \{\omega_1, \dots, \omega_K\}$. The observation vectors $\boldsymbol{x}$ from the N-dimensional space $\mathcal{X}$ (which may be real, discrete or binary)

$$\boldsymbol{x} = (x_1, x_2, \dots, x_N) \in \mathcal{X}, \quad x_n \in \mathcal{X}_n, \quad \mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \dots \times \mathcal{X}_N$$

are supposed to occur randomly according to some class-conditional distributions $P(\boldsymbol{x}|\omega)$ and *a priori* probabilities $p(\omega), \omega \in \Omega$. Recall that, given an observation $\boldsymbol{x} \in \mathcal{X}$, all statistical information about the set of classes Ω is expressed by the Bayes formula for a posteriori probabilities

$$p(\omega|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|\omega)p(\omega)}{P(\boldsymbol{x})}, \quad \boldsymbol{x} \in \mathcal{X} \tag{1}$$

where $P(\boldsymbol{x})$ is the joint unconditional probability distribution of $\boldsymbol{x}$

$$P(\boldsymbol{x}) = \sum_{\omega \in \Omega} P(\boldsymbol{x}|\omega)p(\omega), \quad \boldsymbol{x} \in \mathcal{X}. \tag{2}$$

The posterior distribution $p(\omega|\boldsymbol{x})$ may be used to define a unique final decision by means of the Bayes decision function:[1]

$$d : \mathcal{X} \to \Omega, \quad d(\boldsymbol{x}) = \arg \max_{\omega \in \Omega} \{p(\omega|\boldsymbol{x})\}, \quad \boldsymbol{x} \in \mathcal{X} \tag{3}$$

which is known to minimize the probability of classification error.

In view of Eqs. (1), (3) the statistical solution of a pattern recognition problem can be reduced to estimation of the probabilistic description of classes. Considering PNN we assume that the class-conditional distributions $P(\boldsymbol{x}|\omega)$ may be approximated by finite mixtures

$$P(\boldsymbol{x}|\omega) = \sum_{m \in \mathcal{M}_\omega} F(\boldsymbol{x}|m,\omega) f(m|\omega), \quad \sum_{m \in \mathcal{M}_\omega} f(m|\omega) = 1, \quad f(m|\omega) \geq 0, \quad \omega \in \Omega. \tag{4}$$

Here $f(m|\omega)$ are some probabilistic weights and $\mathcal{M}_\omega$ are the component index sets.

Let us recall that the basic idea of PNN is to view the mixture components as neurons. In this respect the unique correspondence of the components (neurons) to the classes $\omega \in \Omega$ implies an unnatural structural limitation. The parameter ω occurring in the components means that to each class $\omega \in \Omega$ there corresponds a specific subset of neurons or even specific multilayer structure of neurons to be trained separately. On the other hand, one of the most apparent structural properties of ascending neural pathways is a rich branching of axons into multiple endings and, simultaneously, a large number of different axons converging on dendrites of neurons.

In view of the above biological arguments it is more appropriate to define the class-conditional distribution mixtures $P(\boldsymbol{x}|\omega)$ over the same set of components, i. e. we

[1] Possible ties can be solved arbitrarily.

approximate the conditional distributions $P(\boldsymbol{x}|\omega)$ by finite mixtures of components from a given common set. In particular we assume (cf. [7, 6]) that there is a finite set of probability distributions or density functions on $\mathcal{X}$

$$\mathcal{F} = \{F(\cdot|m), m \in \mathcal{M}\} \tag{5}$$

such that each conditional probability distribution $P(\boldsymbol{x}|\omega)$ may be expressed in the form

$$P(\boldsymbol{x}|\omega) = \sum_{m\in\mathcal{M}} F(\boldsymbol{x}|m)f(m|\omega), \quad \omega \in \Omega, \quad \boldsymbol{x} \in \mathcal{X}. \tag{6}$$

Unlike Eq. (4) we assume only one set of components (5) and ignore possible dependence of components on the classes:

$$F(\boldsymbol{x}|m,\omega) = F(\boldsymbol{x}|m), \quad \boldsymbol{x} \in \mathcal{X}, \quad m \in \mathcal{M}, \quad \mathcal{M} \equiv \mathcal{M}_\omega, \quad \omega \in \Omega. \tag{7}$$

In this sense the mixture components $F(\boldsymbol{x}|m)$ may be shared by all class-conditional probability distributions $P(\boldsymbol{x}|\omega)$.

Let us note that the set of component distributions $F(\boldsymbol{x}|m)$ naturally introduces an additional low-level "descriptive" decision problem. By substitution (6) into (2) we can write

$$P(\boldsymbol{x}) = \sum_{m\in\mathcal{M}} F(\boldsymbol{x}|m)f(m), \quad \boldsymbol{x} \in \mathcal{X}, \quad f(m) = \sum_{\omega\in\Omega} f(m|\omega)p(\omega). \tag{8}$$

Here each component in the mixture (8) may correspond to some specific property or situation. Given a vector $\boldsymbol{x} \in \mathcal{X}$, the presence of these properties can be characterized by the conditional probabilities

$$f(m|\boldsymbol{x}) = \frac{F(\boldsymbol{x}|m)f(m)}{P(\boldsymbol{x})}, \quad m \in \mathcal{M}, \quad \boldsymbol{x} \in \mathcal{X}, \quad (P(\boldsymbol{x}) > 0). \tag{9}$$

As it can be seen, there is a simple relation between the a posteriori probabilities of classes and the properties

$$p(\omega|\boldsymbol{x}) = \sum_{m\in\mathcal{M}} p(\omega|m)f(m|\boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{X}, \quad p(\omega|m) = \frac{f(m|\omega)p(\omega)}{f(m)}, \quad \omega \in \Omega. \tag{10}$$

The probabilistic model (6) naturally suggests hybrid optimization of parameters. In particular, only the weights $f(m|\omega)$ have to be estimated in a supervised way to approximate the unknown class conditional distribution mixtures $P(\boldsymbol{x}|\omega)$ whereas the estimation of the mixture components $F(\boldsymbol{x}|m)$ may be non-supervised. The supervised estimation of finite mixtures with shared components is a non-trivial problem which can be solved by a modified EM algorithm in full generality [6].

Let us remark that the assumption of shared components is not restrictive. We can estimate the conditional distributions $P(\boldsymbol{x}|\omega)$ in the form (4) in a supervised way and define the set of shared components $\mathcal{F}$ as a union of class-specific subsets $\mathcal{F}_\omega$. On the other hand, non-supervised estimation of the mixture components

$F(\boldsymbol{x}|m)$ may negatively influence the global performance of the resulting neural network. One can easily verify that the decision information $I(\mathcal{X},\Omega)$ is bounded by the "descriptive" information $I(\mathcal{X},\mathcal{M})$ or in other words that the statistical decision information $I(\mathcal{X},\mathcal{M})$ cannot be increased by creating additional linear combinations of the functions $F(\boldsymbol{x}|m)$ [6, 9]. In this sense reliable estimation of the component distributions $F(\cdot|m) \in \mathcal{F}$ is of primary importance.

Recall that there is a strong similarity between PNN and so called radial basis function (RBF) neural networks. The RBF neural networks usually define the output of a neuron by means of radially symmetrical unimodal function with the aim of multivariate interpolation or approximation of some prescribed output functions (cf. e. g. [7, 21]). However, in case of PNN, we use EM algorithm to compute maximum-likelihood estimates of the class conditional distributions for the sake of Bayesian decision-making. The simplifying assumption of radial symmetry is not necessary in case of mixtures since EM algorithm can be applied to rather general component densities or even to multivariate discrete data.

3. INFORMATION PRESERVING TRANSFORM

In multilayer neural networks each neuron of a given layer formally plays the role of coordinate function of a vector transform $\boldsymbol{T}$ mapping the input space $\mathcal{X}$ into the space of output variables $\mathcal{Y}$. We denote

$$\boldsymbol{T}:\mathcal{X}\to\mathcal{Y}, \quad \mathcal{Y}\subset R^M, \quad \boldsymbol{y}=\boldsymbol{T}(\boldsymbol{x})=(T_1(\boldsymbol{x}),T_2(\boldsymbol{x}),\dots,T_M(\boldsymbol{x}))\in\mathcal{Y}. \tag{11}$$

It has been shown (cf. [4, 7, 28]) that the transform defined in terms of the posterior probabilities $f(m|\boldsymbol{x})$:

$$y_m = T_m(\boldsymbol{x}) = \log f(m|\boldsymbol{x}), \quad \boldsymbol{x}\in\mathcal{X}, \quad m\in\mathcal{M} \tag{12}$$

belongs to a class of information preserving transforms minimizing the entropy of the output space $\mathcal{Y}$. In particular, if we introduce the following notation for the transformed distributions and posterior probabilities

$$Q(\boldsymbol{y}) = P(\boldsymbol{T}^{-1}(\boldsymbol{y})); \quad Q(\boldsymbol{y}|m) = F(\boldsymbol{T}^{-1}(\boldsymbol{y})|m), \quad \boldsymbol{y}\in\mathcal{Y}, \quad m\in\mathcal{M}, \tag{13}$$

$$\boldsymbol{T}^{-1}(\boldsymbol{y}) = \{\boldsymbol{x}\in\mathcal{X}:\boldsymbol{y}=\boldsymbol{T}(\boldsymbol{x})\}, \quad f(m|\boldsymbol{y}) = \frac{Q(\boldsymbol{y}|m)f(m)}{Q(\boldsymbol{y})} \tag{14}$$

and for the related unconditional and conditional Shannon entropies

$$H(\mathcal{M}) = \sum_{m\in\mathcal{M}} -f(m)\log f(m), \quad H(\mathcal{Y}) = \sum_{y\in\mathcal{Y}} -Q(\boldsymbol{y})\log Q(\boldsymbol{y}) \tag{15}$$

$$H(\mathcal{M}|\mathcal{X}) = \sum_{x\in\mathcal{X}} P(\boldsymbol{x})H_x(\mathcal{M}), \quad H_x(\mathcal{M}) = \sum_{m\in\mathcal{M}} -f(m|\boldsymbol{x})\log f(m|\boldsymbol{x})$$

$$H(\mathcal{M}|\mathcal{Y}) = \sum_{y\in\mathcal{Y}} Q(\boldsymbol{y})H_y(\mathcal{M}), \quad H_y(\mathcal{M}) = \sum_{m\in\mathcal{M}} -f(m|\boldsymbol{y})\log f(m|\boldsymbol{y})$$

then we can write

$$I(\mathcal{X},\mathcal{M}) = H(\mathcal{M}) - H(\mathcal{M}|\mathcal{X}) = H(\mathcal{M}) - H(\mathcal{M}|\mathcal{Y}) = I(\mathcal{Y},\mathcal{M}). \tag{16}$$

Roughly speaking, the transform (11), (12) is information preserving as it unifies the points $\boldsymbol{x} \in \mathcal{X}$ with the identical posterior distributions $f(\cdot|\boldsymbol{x})$. For the same reason the decision information $I(\mathcal{X},\Omega)$ is also preserved and we can write

$$I(\mathcal{X},\Omega) = H(\Omega) - H(\Omega|\mathcal{X}) = H(\Omega) - H(\Omega|\mathcal{Y}) = I(\mathcal{Y},\Omega). \tag{17}$$

Simultaneously, it can be shown that the partition of the input space $\mathcal{X}$ induced by the transform $\boldsymbol{T}$ is the "simplest" one in the sense that the entropy $H(\mathcal{Y})$ of the output space $\mathcal{Y}$ is minimized [7, 28]. In Eq. (12) any bijective function could be used but the logarithmic coordinate function is useful since, in case of product components, the output of neuron becomes additive function of input variables [14].

Another argument to use logarithm in Eq. (12) is the fault-tolerant property of the resulting information preserving transform. In particular, if $\boldsymbol{T}$ satisfies the inequality

$$|\,T_m(\boldsymbol{x}) - \ln\left(f(m|\boldsymbol{x}) + f(m)\delta\right)| < \varepsilon, \quad m \in \mathcal{M}, \quad \boldsymbol{x} \in \mathcal{X} \tag{18}$$

for some $\delta > 0$, $\varepsilon > 0$, then the information loss accompanying the transformation $\boldsymbol{T}$ is bounded by the inequality

$$I(\mathcal{X},\mathcal{M}) - I(\mathcal{Y},\mathcal{M}) < \delta + 2\varepsilon. \tag{19}$$

In other words, by using logarithmic function in Eq. (12), we obtain information preserving transform which is fault-tolerant in the sense that a bounded approximation inaccuracy may cause only bounded information loss (cf. [9]).

The principle of information preserving transform is closely related to the papers of Perez [23, 24] and particularly to his concept of ε-sufficiency which has been used by Bialasiewicz to construct ε-sufficient partitioning of the sample space [1]. The constructive method of Bialasiewicz suggests the role of logarithm in the transform (18). Considering general framework of the Bayes decision problem Perez derived upper bounds of the Bayes risk increase caused by a reduction of σ-algebra of the sample space and/or parameter space. In certain sense, the inequality (19) can be viewed as a special case of the general results of Perez.

Let us recall that in practical situations the estimates may differ from the true conditional distributions and therefore the corresponding transform $\boldsymbol{T}$ based on the estimated distributions may be accompanied by some information loss. It is therefore useful that possible consequences of estimation inaccuracy are reasonably bounded by the inequality (19). Moreover, we have shown that the increase of classification error can be partly avoided by combining multiple classifiers [14]. In particular, the underlying information loss connected with the independently designed PNN can be reduced by means of parallel fusion of the corresponding transformation vectors (11). We have derived a hierarchy between different schemes of classifier fusion in terms of information inequalities [15]. In particular, we have shown that the parallel fusion of classifiers is potentially more efficient than various methods based on combining functions. However, the information advantage of parallel fusion is achieved at the expense of the lost simplicity of the combining rules.

4. PROBLEM OF COMPLETE INTERCONNECTION OF PNN

The well known "beauty defect" of the probabilistic approach to neural networks is the biologically unnatural complete interconnection of neurons with all input variables. The complete interconnection requirement has a deep theoretical reason since all class-conditional distributions in the Bayes formula must be normed at the same space and therefore the conditional distributions must be defined at the same set of input variables [17]. We have shown that the resulting structural limitation can be removed in a statistically correct way by a special subspace approach originally proposed in pattern recognition (cf. [5]). In particular, considering PNN we assume the mixture components to be of product form

$$F(\boldsymbol{x}|m) = \prod_{n\in\mathcal{N}} f_n(x_n|m), \quad m \in \mathcal{M}, \quad \mathcal{N} = \{1,\dots,N\} \tag{20}$$

where $f_n(x_n|m)$ are univariate probability distributions or densities. Without leaving the exact framework of Bayesian decision making we introduce additional binary structural parameters $\phi_{mn} \in \{0,1\}$ into the components (20):

$$F(\boldsymbol{x}|m) = \prod_{n\in\mathcal{N}} f_n(x_n|m)^{\phi_{mn}} f_n(x_n|0)^{1-\phi_{mn}}, \qquad \sum_{m\in\mathcal{M}}\sum_{n\in\mathcal{N}} \phi_{mn} = r. \tag{21}$$

By setting the structural parameter $\phi_{mn} = 0$ we can substitute any component-specific density function $f_n(x_n|m)$ by the respective univariate background density $f_n(x_n|0)$ which is common to all components. Thus, the complexity of the resulting mixture simplifies by means of the binary parameters ϕ_{mn} set to zero. We assume the number of the nonzero structural parameters ϕ_{mn} to be given and equal to $r, (r \leq MN)$. We can write the component (21) equivalently as a product

$$F(\boldsymbol{x}|m) = F(\boldsymbol{x}|0)\ G(\boldsymbol{x}|m,\phi_m) \tag{22}$$

where the component functions $G(\boldsymbol{x}|m,\phi_m)$ may be defined on different subspaces

$$G(\boldsymbol{x}|m,\phi_m) = \prod_{n\in\mathcal{N}} \left[\frac{f_n(x_n|m)}{f_n(x_n|0)}\right]^{\phi_{mn}}, \qquad \phi_m = (\phi_{m1},\dots,\phi_{mN}) \in \{0,1\}^N \tag{23}$$

and $F(\boldsymbol{x}|0)$ is a fixed "background" probability density usually defined as a product of unconditional marginals:

$$F(\boldsymbol{x}|0) = \prod_{n\in\mathcal{N}} f_n(x_n|0), \quad f_n(x_n|0) = P_n(x_n). \tag{24}$$

After substitution (22) in Eq. (8) we obtain a modified "structural" distribution mixture

$$P(\boldsymbol{x}) = \sum_{m\in\mathcal{M}} F(\boldsymbol{x}|0)\ G(\boldsymbol{x}|m,\phi_m) f(m). \tag{25}$$

From the point of view of Bayesian decision making the structural mixture (25) is advantageous since the background probability density $F(\boldsymbol{x}|0)$ cancels in the formula of conditional probability (9)

$$f(m|\boldsymbol{x}) = \frac{G(\boldsymbol{x}|m,\phi_m)f(m)}{\sum_{j\in\mathcal{M}} G(\boldsymbol{x}|j,\phi_j)f(j)} \tag{26}$$

and therefore the computation may be confined only to "relevant" variables. In other words the input connections of a single neuron can be confined to an arbitrary subset of input variables (neurons) in a biologically plausible way.

By using the present subspace method we can compute the posterior probabilities of classes $p(\omega|\boldsymbol{x})$ from different subsets of input variables without any preprocessing of the data vectors $\boldsymbol{x} \in \mathcal{X}$. In this sense the structural mixture model (25) represents a statistically correct subspace approach to Bayesian decision-making which is directly applicable to the input space – without any feature selection or dimensionality reduction. Let us recall that in literature the "subspace representation" of classes is usually considered only at the level of extracted features defined as linear combinations of all input variables. For a more detailed discussion cf. [16, 17].

5. STRUCTURAL OPTIMIZATION OF PNN

Numerically the structural mixture model (25) can be optimized by means of EM algorithm in full generality, including the structural parameters (cf. [5, 8, 17]). Given a set of independent observations $\mathcal{S} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_K\}$ we can maximize the corresponding log- likelihood function

$$L = \frac{1}{|\mathcal{S}|}\sum_{x\in\mathcal{S}} \log\left[\sum_{m\in\mathcal{M}} F(\boldsymbol{x}|0)G(\boldsymbol{x}|m,\phi_m)f(m)\right] \tag{27}$$

by means of the modified EM iterations:

$$f(m|\boldsymbol{x}) = \frac{G(\boldsymbol{x}|m,\phi_m)f(m)}{\sum_{j\in\mathcal{M}} G(\boldsymbol{x}|j,\phi_j)f(j)}, \quad m\in\mathcal{M}, \quad \boldsymbol{x}\in\mathcal{S}, \tag{28}$$

$$f^{'}(m) = \frac{1}{|\mathcal{S}|}\sum_{x\in\mathcal{S}} f(m|\boldsymbol{x}), \tag{29}$$

$$f_n^{'}(\cdot|m) = \arg\max_{f_n(\cdot|m)} \left\{\frac{1}{|\mathcal{S}|}\sum_{x\in\mathcal{S}} f(m|\boldsymbol{x})\log f_n(x_n|m)\right\}, \tag{30}$$

$$\gamma_{mn}^{'} = \frac{1}{|\mathcal{S}|}\sum_{x\in\mathcal{S}} f^{(t)}(m|\boldsymbol{x})\log\frac{f_n^{'}(x_n|m)}{f_n(x_n|0)}, \qquad \phi_{mn}^{'} = \begin{cases} 1, & \gamma_{mn}^{'}\in\Gamma_r^{'} \\ 0, & \gamma_{mn}^{'}\notin\Gamma_r^{'}. \end{cases} \tag{31}$$

Here the apostrophe denotes the new iterated values and $\Gamma_r^{'}$ stands for the set of r highest quantities $\gamma_{mn}^{'}$.

The implicit relation (30) can usually be expressed in a closed form. In particular, in case of normal densities

$$f_n(x_n|\mu_{mn},\sigma_{mn}) = \frac{1}{\sqrt{(2\pi)}\sigma_{mn}}\exp\left\{-\frac{1}{2}\frac{(x_n-\mu_{mn})^2}{\sigma_{mn}^2}\right\} \tag{32}$$

Eq. (30) has the following solution

$$\mu'_{mn} = \frac{1}{\sum_{x\in\mathcal{S}} f(m|\boldsymbol{x})} \sum_{x\in\mathcal{S}} x_n f(m|\boldsymbol{x}), \tag{33}$$

$$\left(\sigma'_{mn}\right)^2 = \frac{1}{\sum_{x\in\mathcal{S}} f(m|\boldsymbol{x})} \sum_{x\in\mathcal{S}} \left(x_n^2 - \mu'_{mn}\right)^2 f(m|\boldsymbol{x}) \tag{34}$$

and the structural criterion has the form

$$\gamma'_{mn} = f'(m)\left(\frac{(\mu'_{mn}-\mu_{0n})^2}{\sigma_{0n}^2} + \left(\frac{\sigma'_{mn}}{\sigma_{0n}^2}\right)^2 - \log\left(\frac{\sigma'_{mn}}{\sigma_{0n}}\right)^2 - 1\right). \tag{35}$$

Similarly, in case of discrete distributions

$$f_n(x_n|m) \geq 0,\ x_n \in \mathcal{X}_n, \qquad \sum_{x_n\in\mathcal{X}_n} f_n(x_n|m) = 1, \qquad m \in \mathcal{M}$$

we can write explicit solution of Eq. (30) and the structural criterion in the form

$$f'_n(\xi|m) = \frac{1}{\sum_{x\in\mathcal{S}} f(m|\boldsymbol{x})} \sum_{x\in\mathcal{S}} \delta(\xi, x_n) f(m|\boldsymbol{x}), \quad \xi \in \mathcal{X}_n, \tag{36}$$

$$\gamma'_{mn} = \frac{1}{|\mathcal{S}|} \sum_{x\in\mathcal{S}} f(m|\boldsymbol{x}) \log \frac{f'_n(x_n|m)}{f_n(x_n|0)}. \tag{37}$$

A detailed discussion of the monotone convergence of "structural" EM algorithm can be found in [5] and [8].

Let us note that arbitrary weighting of the training data set $\mathcal{S}$ can be equivalently realized by repeating the data vectors in $\mathcal{S}$. Denoting $\lambda(\boldsymbol{x})$ a weight function we can write directly the weighted version of the likelihood function [17]:

$$L_\lambda = \frac{1}{|\mathcal{S}|} \sum_{x\in\mathcal{S}} \lambda(\boldsymbol{x}) \log\left[\sum_{m\in\mathcal{M}} F(\boldsymbol{x}|0) G(\boldsymbol{x}|m, \phi_m) f(m)\right] \tag{38}$$

$$\tag{39}$$

and also the corresponding modification of the structural EM algorithm (cf. [17]):

$$f'(m) = \frac{1}{\Lambda(\mathcal{S})} \sum_{x\in\mathcal{S}} \lambda(\boldsymbol{x}) f(m|\,\boldsymbol{x}), \qquad \Lambda(\mathcal{S}) = \sum_{x\in\mathcal{S}} \lambda(\boldsymbol{x}), \tag{40}$$

$$f'_n(\cdot|m) = \arg\max_{f_n(\cdot|m)} \left\{\frac{1}{\Lambda(\mathcal{S})} \sum_{x\in\mathcal{S}} \lambda(\boldsymbol{x}) f(m|\boldsymbol{x}) \log f_n(x_n|m)\right\}, \tag{41}$$

$$\gamma'_{mn} = \frac{1}{\Lambda(\mathcal{S})} \sum_{x\in\mathcal{S}} \lambda(\boldsymbol{x}) f(m|\boldsymbol{x}) \log \frac{f'_n(x_n|m)}{f_n(x_n|0)}. \tag{42}$$

Recently we have shown [18] for discrete space $\mathcal{X}$ that if $P^*(\boldsymbol{x})$ is the true distribution and $\lambda(\boldsymbol{x})$ is a positive bounded function on $\mathcal{X}$ then the maximization of the weighted likelihood function L_λ (cf. (38)) is asymptotically equivalent to maximum-likelihood estimation of the distribution $\tilde{P}(\boldsymbol{x})$ obtained by analogous weighting of the original distribution $P^*(\boldsymbol{x})$:

$$\tilde{P}(\boldsymbol{x}) = \frac{\lambda(\boldsymbol{x})}{\Lambda^*} P^*(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{X}, \qquad \Lambda^* = \sum_{x \in \mathcal{X}} \lambda(\boldsymbol{x}) P^*(\boldsymbol{x}). \tag{43}$$

Here Λ^* is a norming coefficient. From the above assertion it follows that, for the sample-size approaching infinity, the posterior probabilities

$$f^*(m|\boldsymbol{x}) = \frac{F^*(\boldsymbol{x}|m) f^*(m)}{P^*(\boldsymbol{x})}, \quad m \in \mathcal{M}, \quad \boldsymbol{x} \in \mathcal{X} \tag{44}$$

are asymptotically invariant with respect to weighting of the distribution $P^*(\boldsymbol{x})$ by $\lambda(\boldsymbol{x})$. In particular, we can write

$$\tilde{P}(\boldsymbol{x}) = \sum_{m \in \mathcal{M}} \tilde{F}(\boldsymbol{x}|m) \tilde{f}(m) = \frac{\lambda(\boldsymbol{x})}{\Lambda^*} P^*(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathcal{X} \tag{45}$$

with the components satisfying Eqs.

$$\tilde{F}(\boldsymbol{x}|m) = \frac{\lambda(\boldsymbol{x})}{\Lambda^*_m} F^*(\boldsymbol{x}|m), \quad \Lambda^*_m = \sum_{x \in \mathcal{X}} \lambda(\boldsymbol{x}) F^*(\boldsymbol{x}|m), \quad \tilde{f}(m) = \frac{\Lambda^*_m}{\Lambda^*} f^*(m), \tag{46}$$

and therefore, by substitution, we obtain

$$\tilde{f}(m|\boldsymbol{x}) = \frac{\tilde{F}(\boldsymbol{x}|m)\tilde{f}(m)}{\tilde{P}(\boldsymbol{x})} = f^*(m|\boldsymbol{x}), \quad m \in \mathcal{M}. \tag{47}$$

In this sense the design of information preserving transform is asymptotically invariant with respect to arbitrary weighting of training data by a positive bounded function $\lambda(\boldsymbol{x})$.

6. SEQUENTIAL MODIFICATION OF EM ALGORITHM

An important aspect of biological interpretation is the possibility of a sequential optimization of parameters. In the original form the EM algorithm is a typical off-line estimation method using all training data at each iteration. However, assuming an infinite data sequence

$$\mathcal{S} = \{\boldsymbol{x}^{(t)}\}_{t=1}^{\infty}, \quad \boldsymbol{x}^{(t)} \in \mathcal{X}, \tag{48}$$

we can write the EM algorithm in a modified sequential form which corresponds to a single “infinite” iteration with intermediate updating of the estimated parameters (cf. [10, 13, 19]).

Considering a sequential version of EM algorithm we assume some fixed structural parameters ϕ_{mn} since "on-line" structural optimization does not seem to have an efficient counterpart in biological neural systems. The adaptive properties of neural systems are mainly enabled by the plasticity of existing synapses while the structure of interconnections can be assumed to be given and essentially fixed in a particular case. It is supposed that the structural properties of biological neural systems have been optimized during the long process of phylogenetic evolution.

For the sake of a sequential modification of EM algorithm we define first the index of periodically substituted parameters by Eq.

$$k_t = [:\alpha \log t:], \quad k_t \in \{0,1,2,\dots\}, \quad t = 1,2,\dots \tag{49}$$

where [: :] denotes integer part of the parenthesized expression, i. e. k_t is a step-function of t. In the following the parameter estimates $\hat{f}^{(t)}(m), \hat{f}_n^{(t)}(\cdot|m)$ are sequentially computed but the new parameter values are substituted only periodically – at the points of change of the substitution index k_t:

$$k_t > k_{t-1} \;\Rightarrow\quad f^{(k_t)}(m) = \hat{f}^{(t)}(m), \qquad f^{(k_t)}(\cdot|m) = \hat{f}^{(t)}(\cdot|m), \tag{50}$$

In this sense $G^{(k)}(\cdot|m,\phi_m)$ denotes the component function $G(\cdot|m,\phi_m)$ with the kth parameter set substituted. The sequential EM procedure can be described as follows:

$$f^{(k_t)}(m|\boldsymbol{x}^{(t)}) = \frac{G^{(k_t)}(\boldsymbol{x}^{(t)}|m,\phi_m)f^{(k_t)}(m)}{\sum_{j\in\mathcal{M}} G^{(k_t)}(\boldsymbol{x}^{(t)}|j,\phi_j)f^{(k_t)}(j)}, \tag{51}$$

$$\hat{f}^{(t+1)}(m) = \frac{1}{t}\sum_{\tau=1}^{t} f^{(k_\tau)}(m|\boldsymbol{x}^{(\tau)}), \tag{52}$$

$$\hat{f}_n^{(t+1)}(\cdot|m) = \arg\max_{f_n(\cdot|m)} \left\{ \frac{1}{t}\sum_{\tau=1}^{t} f^{(k_\tau)}(m|\boldsymbol{x}^{(\tau)}) \log f_n(x_n^{(\tau)}|m) \right\}. \tag{53}$$

Again, in analogy with the Eqs. (33) – (36), we can write solution of Eq. (53) in explicit form. In particular, we obtain for Gaussian densities the following sequential Eqs.

$$\mu_{mn}^{(k_t)}(m) = \hat{\mu}_{mn}^{(t)}(m), \qquad \sigma_{mn}^{(k_t)} = \hat{\sigma}_{mn}^{(t)}, \tag{54}$$

$$\hat{\mu}_{mn}^{(t+1)} = \frac{1}{\sum_{\tau=1}^{t} f^{(k_\tau)}(m|\boldsymbol{x}^{(\tau)})} \sum_{\tau=1}^{t} x_n^{(\tau)} f^{(k_\tau)}(m|\boldsymbol{x}^{(\tau)}), \tag{55}$$

$$(\hat{\sigma}_{mn}^{(t+1)})^2 = \frac{1}{\sum_{\tau=1}^{t} f^{(k_\tau)}(m|\boldsymbol{x}^{(\tau)})} \sum_{\tau=1}^{t} \left(x_n^{(\tau)} - \mu_{mn}^{(k_\tau)}\right)^2 f^{(k_\tau)}(m|\boldsymbol{x}^{(\tau)}) \tag{56}$$

and similarly, in case of discrete distributions, we can write

$$\hat{f}_n^{(t+1)}(\xi|m) = \frac{1}{\sum_{\tau=1}^{t} f^{(k_\tau)}(m|\boldsymbol{x}^{(\tau)})} \sum_{\tau=1}^{t} \delta(\xi, x_n) f^{(k_\tau)}(m|\boldsymbol{x}^{(\tau)}), \quad \xi \in \mathcal{X}_n. \tag{57}$$

Note that, introducing notation

$$\gamma(t) = \frac{f^{(k_t)}(m|\boldsymbol{x}^{(t)})}{\sum_{\tau=1}^{t} f^{(k_\tau)}(m|\boldsymbol{x}^{(\tau)})}, \quad t = 1, 2, \dots \tag{58}$$

we can express the parameter estimates (52), (55) – (57) in the following more intuitive way

$$\hat{f}^{(t+1)}(m) = \left(1 - \frac{1}{t}\right) \hat{f}^{(t)}(m) + \frac{1}{t} f^{(k_t)}(m|\boldsymbol{x}^{(t)}), \tag{59}$$

$$\hat{\mu}_{mn}^{(t+1)} = (1 - \gamma(t))\, \hat{\mu}_{mn}^{(t)} + \gamma(t) x_n^{(t)}, \tag{60}$$

$$\left(\hat{\sigma}_{mn}^{(t+1)}\right)^2 = (1 - \gamma(t)) \left(\hat{\sigma}_{mn}^{(t)}\right)^2 + \gamma(t) \left(x_n^{(t)} - \mu_{mn}^{(k_t)}\right)^2, \tag{61}$$

$$\hat{f}_n^{(t+1)}(\xi|m) = (1 - \gamma(t))\, \hat{f}_n^{(t)}(\xi|m) + \gamma(t) \delta(\xi, x_n^{(t)}), \quad \xi \in \mathcal{X}_n. \tag{62}$$

Let t_k be a point of the kth parameter substitution (kth upgrade). Then the difference $T_k = t_k - t_{k-1}$ represents the interval between the two consecutive substitutions t_{k-1}, t_k. In case of definition (49) the length of the "substitution interval" T_k increases approximately by a coefficient $\exp\{1/\alpha\}$, i. e. we have

$$\frac{T_{k+1}}{T_k} = \frac{(t_{k+1} - t_k)}{(t_k - t_{k-1})} \doteq \exp\{1/\alpha\}, \quad k = 1, 2, \dots \tag{63}$$

In a recent paper [13] we have applied the sequential EM algorithm to binary data in order to demonstrate strictly modular properties of PNN. In numerical experiments we have observed that the coefficient $\exp\{1/\alpha\}$ chosen from the interval $\langle 1.2, 1.4 \rangle$ yields good results without essential differences [13]. It should be noted that the above sequential scheme does not guarantee the basic monotonic property of EM algorithm. However, in some cases, the sequential scheme yields even better results than the standard EM procedure because the underlying sequential computation includes much longer data sequence than the standard EM iteration procedure bounded to a given finite set (cf. [13]).

Let us mention finally a biological analogy of the above periodical substitutions arising in the sequential version of EM algorithm. From a neurophysiological point of view it is generally assumed that the neural activity specifically influences the concentration of some chemical stuffs or energetic balance of neurons. As a consequence some metabolical changes or growth processes responsible for adaptation may follow. In this sense some delay can be expected between the primary specific activity of a neuron and the corresponding adaptive changes. In this sense the periodical substitutions in the sequential EM scheme could correspond to the sleep phase which is known to be essential for the long term memory of biological neural systems.

7. NEUROPHYSIOLOGICAL INTERPRETATION

Since very beginnings the research of neural networks is motivated by wonderful performance of biological neural networks like e. g. mammalian central neural system. On the other hand, its elementary units – neurons are known to be relatively unreliable and inaccurate. It is therefore assumed that the excellent properties of neural systems have to be based on some very efficient and robust principles. In view of these facts the main motivation of PNN is to explain the basic functional properties of neurons theoretically in the framework of a general statistical decision problem.

From the neurophysiological point of view the conditional probability $f(m|\boldsymbol{x})$ can be naturally interpreted as a measure of excitation or probability of firing of the mth neuron given the input pattern $\boldsymbol{x} \in \mathcal{X}$. The output signal of the mth neuron y_m is defined as logarithm of the excitation $f(m|\boldsymbol{x})$ and therefore logarithm plays the role of activation function or response curve. In view of Eqs. (23), (26) we can write

$$y_m = T_m(\boldsymbol{x}) = \log f(m|\boldsymbol{x})$$

$$= \quad \log f(m) \ + \sum_{n\in\mathcal{N}} \phi_{mn} \log \frac{f_n(x_n|m)}{f_n(x_n|0)} \ - \ \log[\sum_{j\in\mathcal{M}} G(\boldsymbol{x}|j,\phi_j)f(j)]. \tag{64}$$

Consequently, we may assume the first term on the right-hand side of Eq. (64) to be responsible for spontaneous activity of the mth neuron. The second term in Eq. (64) summarizes contributions of input variables x_n (neurons) of the subset defined by means of binary structural parameters $\phi_{mn} = 1$. In this sense, the term

$$w_{mn}(x_n) = \log f_n(x_n|m) - \log f_n(x_n|0) \tag{65}$$

can be viewed as the current synaptic weight of the nth neuron at input of the mth neuron – as a function of the input value x_n. The effectiveness of the synaptic transmission, as expressed by the formula (65), combines the statistical properties of the input variable x_n with the activity of the "postsynaptic" neuron "m". In words, the synaptic weight (65) is high when the input signal x_n frequently takes part in excitation of the mth neuron and, in turn, it is low when the input signal x_n usually doesn't contribute to the excitation of the mth neuron. This formulation resembles the classical Hebb's postulate of learning (cf. Hebb (1949), p. 62):

"When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic changes take place in one or both cells such that A's efficiency as one of the cells firing B, is increased."

The last term in (64) includes the norming coefficient responsible for competitive properties of neurons and therefore it can be interpreted as a cumulative effect of special neural structures performing lateral inhibition. This term is identical for all components of the underlying mixture and, for this reason, the Bayesian decision-making would not be influenced by its accuracy. However, in case of hidden-layer neurons, there is a problem of lateral inhibition accuracy. In case of information preserving transform the lateral inhibition should summarize the output signals of exactly those neurons to which it is applied. However, in biological neural systems such an exact correspondence is hardly possible.

Let us recall in this connection the invariance of information preserving transform with respect to weighting (cf. Section 5). If we assume an incorrect correspondence of neurons in the lateral inhibition structure then the resulting effect of incorrect norming can be represented by a weighting function:

$$\tilde{f}(m|\boldsymbol{x}) = \frac{G(\boldsymbol{x}|m,\phi_m)f(m)}{\sum_{j\in\tilde{\mathcal{M}}} G(\boldsymbol{x}|j,\phi_j)f(j)} = \frac{\sum_{j\in\mathcal{M}} G(\boldsymbol{x}|j,\phi_j)f(j)}{\sum_{j\in\tilde{\mathcal{M}}} G(\boldsymbol{x}|j,\phi_j)f(j)} f(m|\boldsymbol{x}) = \tilde{\lambda}(\boldsymbol{x})f(m|\boldsymbol{x}).$$

Here $\tilde{\mathcal{M}}$ denotes the incorrect set of neurons. In this sense the effect of incorrect lateral inhibition would be counter balanced by the asymptotic invariance of EM-learning with respect to weighting.

Let us also note that, in the exact form, mathematical expectation of the last “norming” term in (64) can be expressed by means of information-divergence

$$I(P(\cdot)||F(\cdot|0)) = E_P\left\{\log\frac{P(\boldsymbol{x})}{F(\boldsymbol{x}|0)}\right\} = E_P\left\{\log\frac{P(\boldsymbol{x})}{\prod_{n\in\mathcal{N}} P_n(x_n)}\right\}. \qquad (66)$$

The last formula has been proposed by Perez [24] as a measure of dependence of the involved variables. Perez suggested to call the quantity (66) the “dependence tightness” as it is nonnegative and for independent variables it is zero. In view of Eq. (66) the effect of lateral inhibition should be high for highly correlated inputs and low or even zero for nearly independent input signals.

8. CONCLUSION

The functional understanding of biological neural networks is known to be difficult because of many highly specific features. Considering PNN we should mention at least three major biological interpretation problems. First, the information preserving transform presumes a real output signal of neurons which are known to produce binary “spikes”. Obviously a real output value may correspond to the output frequency of spikes but the evaluation mechanism of spike frequency in lateral inhibition and in the learning processes may be rather complex and remains unclear. The second problem relates to fact that the functional properties of probabilistic neurons correspond to “long-term” parameter estimates computed by EM algorithm. The present PNN cannot explain the well-known short-term synaptical plasticity and the related processes. Finally there is a problem of recurrent processes which represent one of the most apparent but not well understood feature of the biological neural networks.

On the other hand there are strong arguments to support PNN as a statistical model of information processing in biological neural networks: The principle of PNN derives from a general well justified statistical method of pattern recognition. The functioning of multilayer PNN is “guaranteed” by means of information preserving transform. The role of logarithm as activation function of probabilistic neuron is closely related to the concept of ε-sufficiency introduced by Perez. In the feedforward structures each neuron may include arbitrary subset of lower-level output variables as input space. The underlying structural mixture model can be optimized by means of EM algorithm in full generality. The structural PNN can be designed by a sequential version of EM algorithm allowing for sequential processing of training data. Finally, the probabilistic neuron can be interpreted in neurophysiological terms up to the level of functional properties of biological neurons. In this way we can find possible theoretical counterparts of the well known neurophysiological properties. In particular, the general formula for synaptical weights can be viewed as a theoretical justification of the well known Hebbian principle of learning. Similarly, the mean effect of lateral inhibition can be viewed as the dependence tightness of involved variables – the term proposed by Perez in connection with dependence structure simplification.

ACKNOWLEDGEMENT

This research was supported by the Czech Science Foundation under grant No. 102/07/1594 and partially by the Ministry of Education, Youth and Sports of the Czech Republic under projects 1M0572 DAR, 2C06019 ZIMOLEZ and by the EC project No. FP6-507752 MUSCLE.

(Received July 11, 2006.)

REFERENCES

[1] J. Bialasiewicz: Statistical data reduction via construction of sample space partitions. Kybernetika *6* (1970), 6, 371–379.

[2] A. P. Dempster, N. M. Laird, and D. B. Rubin: Maximum-likelihood from incomplete data via the EM algorithm. J. Royal Statist. Soc. *B 39* (1977), 1–38.

[3] J. Grim: On numerical evaluation of maximum-likelihood estimates for finite mixtures of distributions. Kybernetika *18* (1982), 3, 173–190.

[4] J. Grim: Design and optimization of multilevel homogeneous structures for multivariate pattern recognition. In: Fourth FORMATOR Symposium 1982, Academia, Prague 1982, pp. 233–240.

[5] J. Grim: Multivariate statistical pattern recognition with non-reduced dimensionality. Kybernetika *22* (1986), 6, 142–157.

[6] J. Grim: Maximum-likelihood design of layered neural networks. In: Proc. Internat. Conference Pattern Recognition. IEEE Computer Society Press, Los Alamitos 1996, pp. 85–89.

[7] J. Grim: Design of multilayer neural networks by information preserving transforms. In: Third European Congress on Systems Science (E. Pessa, M. P. Penna, and A. Montesanto, eds.). Edizioni Kappa, Roma 1996, pp. 977–982.

[8] J. Grim: Information approach to structural optimization of probabilistic neural networks. In: Fourth European Congress on Systems Science (L. Ferrer and A. Caselles, eds.). SESGE, Valencia 1999, pp. 527–539.

[9] J. Grim: Discretization of probabilistic neural networks with bounded information loss. In: Computer–Intensive Methods in Control and Data Processing. (Preprints of the 3rd European IEEE Workshop CMP'98, Prague 1998, J. Rojicek et al., eds.), ÚTIA AV ČR, Prague 1998, pp. 205–210.

[10] J. Grim: A sequential modification of EM algorithm. In: Proc. Classification in the Information Age (W. Gaul and H. Locarek-Junge, eds., Studies in Classification, Data Analysis, and Knowledge Organization), Springer, Berlin 1999, pp. 163–170.

[11] J. Grim J.: Self-organizing maps and probabilistic neural networks. Neural Network World *10* (2000), 3, 407–415.

[12] J. Grim: Probabilistic Neural Networks (in Czech). In: Umělá inteligence IV. (V. Mařík, O. Štěpánková, and J. Lažanský, eds.), Academia, Praha 2003, pp. 276–312.

[13] J. Grim, P. Just, and P. Pudil: Strictly modular probabilistic neural networks for pattern recognition. Neural Network World *13* (2003), 6, 599–615.

[14] J. Grim, J. Kittler, P. Pudil, and P. Somol: Combining multiple classifiers in probabilistic neural networks. In: Multiple Classifier Systems (Lecture Notes in Computer Science 1857, J. Kittler and F. Roli, eds.). Springer, Berlin 2000, pp. 157–166.

[15] J. Grim, J. Kittler, P. Pudil, and P. Somol: Information analysis of multiple classifier fusion. In: Multiple Classifier Systems 2001 (Lecture Notes in Computer Science 2096, J. Kittler and F. Roli, eds.). Springer, Berlin – New York 2001, pp. 168–177.

[16] J. Grim, J. Kittler, P. Pudil, and P. Somol: Multiple classifier fusion in probabilistic neural networks. Pattern Analysis & Applications *5* (2002), 7, 221–233.

[17] J. Grim, P. Pudil, and P. Somol: Recognition of handwritten numerals by structural probabilistic neural networks. In: Proc. Second ICSC Symposium on Neural Computation (H. Bothe and R. Rojas, eds.). ICSC, Wetaskiwin 2000, pp. 528–534.

[18] J. Grim, P. Pudil, and P. Somol: Boosting in probabilistic neural networks. In: Proc. 16th International Conference on Pattern Recognition (R. Kasturi, D. Laurendeau and C. Suen, eds.). IEEE Computer Society, Los Alamitos 2002, pp. 136–139.

[19] J. Grim, P. Somol, P. Pudil, and P. Just: Probabilistic neural network playing a simple game. In: Artificial Neural Networks in Pattern Recognition (S. Marinai and M. Gori, eds.). University of Florence, Florence 2003, pp. 132–138.

[20] J. Grim, P. Somol, and P. Pudil: Probabilistic neural network playing and learning Tic-Tac-Toe. Pattern Recognition Letters, Special Issue *26* (2005), 12, 1866–1873.

[21] S. Haykin: Neural Networks: A Comprehensive Foundation. Morgan Kaufman, San Mateo 1993.

[22] G. J. McLachlan and D. Peel: Finite Mixture Models. Wiley, New York – Toronto 2000.

[23] A. Perez: Information, ε-sufficiency and data reduction problems. Kybernetika *1* (1965), 4, 297–323.

[24] A. Perez: ε-admissible simplification of the dependence structure of a set of random variables. Kybernetika *13* (1977), 6, 439–449.

[25] M. I. Schlesinger: Relation between learning and self-learning in pattern recognition (in Russian). Kibernetika (1968), 6, 81–88.

[26] D. F. Specht: Probabilistic neural networks for classification, mapping or associative memory. In: Proc. IEEE Internat. Conference on Neural Networks 1988, Vol. I, pp. 525–532.

[27] L. R. Streit and T. E. Luginbuhl: Maximum-likelihood training of probabilistic neural networks. IEEE Trans. Neural Networks *5* (1994), 764–783.

[28] I. Vajda and J. Grim: About the maximum information and maximum likelihood principles in neural networks. Kybernetika *34* (1998), 4, 485–494.

[29] S. Watanabe and K. Fukumizu: Probabilistic design of layered neural networks based on their unified framework. IEEE Trans. Neural Networks *6* (1995), 3, 691–702.

[30] L. Xu and M. I. Jordan: On convergence properties of the EM algorithm for Gaussian mixtures. Neural Computation *8* (1996), 129–151.

Jiří Grim, Institute of Information Theory and Automation – Academy of Sciences of the Czech Republic, Pod Vodárenskou věží 4, 182 08 Praha 8. Czech Republic.
e-mail: grim@utia.cas.cz