KYBERNETIKA — VOLUME *34* (1998), NUMBER 5, PAGES 497 – 514

NEURAL NETWORK REALIZATIONS OF BAYES DECISION RULES FOR EXPONENTIALLY DISTRIBUTED DATA[1]

Igor Vajda, Belomír Lonek, Viktor Nikolov and Arnošt Veselý

For general Bayes decision rules there are considered perceptron approximations based on sufficient statistics inputs. A particular attention is paid to Bayes discrimination and classification. In the case of exponentially distributed data with known model it is shown that a perceptron with one hidden layer is sufficient and the learning is restricted to synaptic weights of the output neuron. If only the dimension of the exponential model is known, then the number of hidden layers will increase by one and also the synaptic weights of neurons from both hidden layers have to be learned.

1. INTRODUCTION

We consider random observations $\boldsymbol{x}$ distributed on R^n and suppose that real valued actions (decisions) are undertaken on the basis of these observations. Then the Bayes decision rule $\delta^*(\boldsymbol{x})$ is a real-valued function defined on R^n. It is known (see e. g. Sec. 6 in Müller et al [12]) that every reasonable mapping $R^n \to R$, and consequently every reasonable Bayes rule $\delta^*(\boldsymbol{x})$, can be approximated by a perceptron with the input $\boldsymbol{x}$, consisting of at most two hidden layers of neurons and one output neuron. The well-known learning by error back-propagation asymptotically leads to consistent estimates of unknown synaptic weights of all neurons under consideration.

Unfortunately, if the dimension of the input $\boldsymbol{x}$ is very large then the extent of iterative learning steps needed to obtain weight estimates of desired precision is not practically achievable (cf. the learning procedures for perceptrons in Sec. 6 of [12]). Very large dimensions of inputs are typical when observations are taken on random processes.

One possibility to keep the dimensionality under control is to replace the observations $\boldsymbol{x} = (x_1, \ldots, x_n)$ by their "sufficiently representative" features $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_k)$. It is known (cf. e. g. Devijver and Kittler [4], Berger [1], Bock [3], Vajda and Grim [15]) that if the features $\boldsymbol{\phi}(\boldsymbol{x}) = (\phi_1(\boldsymbol{x}), \ldots, \phi_k(\boldsymbol{x})) \in R^k$ defined for all $\boldsymbol{x} \in R^n$ contain all relevant information about $\boldsymbol{x}$ then there exists a mapping $\delta : R^k \to R$

[1] Supported by the Grant Agency of the Czech Republic under grant 102/94/0320 and by the Grant Agency of the Academy of Sciences of the Czech Republic under grant 2075703.

one-one related to δ^* in the sense

$$\delta(\boldsymbol{\phi}(\boldsymbol{x})) = \delta^*(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in R^n. \tag{1}$$

This means that δ^* can be approximated by a perceptron of the above considered type with the inputs $\boldsymbol{\phi}(\boldsymbol{x}) \in R^k$. If $k \ll n$ then the new perceptron is essentially simpler than the original one.

The last paragraph describes the main idea of present paper, outlined already in Vajda [14]. We consider decision problems for observations $\boldsymbol{x}$ with information-preserving features $\boldsymbol{\phi}(\boldsymbol{x})$ of a dimension $k < n$. Perceptrons of the above considered type are then used to approximate the Bayes version δ defined on the feature space R^k.

We restrict ourselves to observations $\boldsymbol{x}$ exponentially distributed, with an unknown parameter $\boldsymbol{\theta}$ from a space $\Theta \subset R^m$ of known dimension $m < n$. Then the maximum likelihood estimator (MLE) $\hat{\boldsymbol{\theta}}(\boldsymbol{x})$ of $\boldsymbol{\theta}$ takes on values in R^m and contains all relevant information about observations $\boldsymbol{x}$ (it is the so called sufficient statistics, cf. e. g. Brown [2]). Thus if the MLE is known (which takes place if the exponential family is known) then one can take $k = m$ and $\boldsymbol{\phi}(\boldsymbol{x}) = \hat{\boldsymbol{\theta}}(\boldsymbol{x})$. Practically all families used for stochastic modelling of independent observations are exponential (Bernoulli, Pascal, Poisson, Maxwell, Rayleigh, Pareto, Student, chi-square, F, etc.). Multinomial, multivariate normal, and many other models of dependent observations used in biology, medicine, image and speech processing, telecomunications, stock market analysis etc., are exponential, including observations on all common models of random processes (see Küchler and Sørensen [9]).

Main attention of this paper is focused on the most simple type of decision problem which is discrimination (classification), characterized by binary (M-ary) parameter and decision spaces. We prove that the Bayes discrimination function $\delta^*(\boldsymbol{x})$ coincides with the response of a neural network with input $\boldsymbol{x}$, consisting of one hidden layer of $m + 3$ units with responses explicitely specified by the exponential model, and one output neuron. All $m+3$ weights of the output neuron are unknown unless the distributions of discriminated observations are given a priori. If the exponential model itself is a priori unknown, then we show that the Bayes discrimination function can be approximated by a perceptron with input $\boldsymbol{x}$, two hidden layers of neurons and one output neuron. The weights of all neurons can be learned by the error back-propagation. It is shown that this learning procedure is in some cases computationally feasible. Extensions to the Bayes classification are discussed too.

2. BAYES RULES

Let the probability distribution P of an observation $\boldsymbol{x} = (x_1, \dots, x_n)$ be from a family $\mathcal{P} = \{P_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \Theta\}$ of probability distributions on R^n with densities $\{p_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \Theta\}$ with respect to a σ-finite measure μ (Lebesgue measure if the distributions are continuous, counting measure if the distributions has an at most countable support in of R^n). The parameter space Θ is supposed to be a subset of R^m.

Let A be a set of possible actions, $L(\boldsymbol{\theta}, a)$ a nonnegative loss function defined on $\Theta \times A$, and $\pi(\boldsymbol{\theta})$ a probability density on Θ with respect to a σ-finite measure ν on

R^m. For every decision rule $\delta^* : R^n \to A$

$$B(\delta^*) = \int_{R^n} \int_{R^m} L(\boldsymbol{\theta}, \delta^*(\boldsymbol{x}))\, p_{\boldsymbol{\theta}}(\boldsymbol{x})\, \mathrm{d}\mu(\boldsymbol{x})\, \pi(\theta)\, \mathrm{d}\nu(\boldsymbol{\theta})$$

is the Bayes risk with respect to the prior distribution π. If for all $\boldsymbol{x} \in R^n$

$$\int_{R^m} L(\boldsymbol{\theta}, \delta^*(\boldsymbol{x}))\, p_{\boldsymbol{\theta}}(\boldsymbol{x})\, \pi(\theta)\, \mathrm{d}\nu(\boldsymbol{\theta}) = \inf_{a \in A} \int_{R^m} L(\boldsymbol{\theta}, a)\, p_{\boldsymbol{\theta}}(\boldsymbol{x})\, \pi(\boldsymbol{\theta})\, \mathrm{d}\nu(\boldsymbol{\theta}) \tag{2}$$

then δ^* is the Bayes rule that minimizes the Bayes risk.

The following result is not new (cf. e.g. Berger [1]). It is presented here, together with a simple proof, for later references.

Assertion 1. If there exists a statistic $\boldsymbol{\phi} : R^n \to R^k$ sufficient for the family $\mathcal{P}$ then the Bayes rule $\delta^* : R^n \to A$ satisfies (1) for a Bayes version $\delta : R^k \to A$.

Proof. Let δ^* be a Bayes decision rule and $\boldsymbol{\phi} : R^n \to R^k$ a mapping sufficient for the family $\mathcal{P}$. By the factorization theorem of mathematical statistics, this means nothing but the existence of functions $\{g_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \Theta\}$ defined on R^k and h defined on R^n such that

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = g_{\boldsymbol{\theta}}(\boldsymbol{\phi}(\boldsymbol{x}))\, h(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x} \in R^n.$$

It follows from here and from (2) that $\delta(\boldsymbol{x})$ depends on $\boldsymbol{x}$ only through the value $\boldsymbol{\phi}(\boldsymbol{x})$, i.e. there exists $\delta : R^k \to A$ satisfying (1). □

In this paper we are interested in the problem under what assumptions about decision problems the Bayes rules $\delta^*(\boldsymbol{x})$ can be realized as responses of perceptrons, and under what assumptions the unknown synaptic weights of these perceptrons can be learned in a "reasonable time".

By a *neural network* with input $\boldsymbol{x}$ and one hidden layer we mean a triplet $\langle M, \boldsymbol{s}, \boldsymbol{w}\rangle$ where M defined the number of hidden units, $\boldsymbol{s} = (s_1, \dots, s_M)$ is an R^M-valued function of the input defining responses of the hidden units $1, \dots, M$, and $\boldsymbol{w} = (w_1, \dots, w_M) \in R^M$ are synaptic weights of the output neuron of the perceptron. This means that the response of this neuron is $\varphi(\boldsymbol{s} \cdot \boldsymbol{w})$ provided φ is the activating functions and $\cdot$ denotes the scalar product. E.g. for the sigmoidal activating function $\varphi_\beta(h) = (1 + e^{\beta h})^{-1}$ one has

$$\varphi_\beta(\boldsymbol{s} \cdot \boldsymbol{w}) = \left(1 + e^{\beta\, \boldsymbol{s} \cdot \boldsymbol{w}}\right)^{-1} \quad \text{and} \quad \lim_{\beta \to \infty} \varphi_\beta(\boldsymbol{s} \cdot \boldsymbol{w}) = 1_{(-\infty,0)}(\boldsymbol{s} \cdot \boldsymbol{w}) \quad \text{for} \quad \boldsymbol{s} \cdot \boldsymbol{w} \neq 0. \tag{3}$$

Obviously, the response of the whole network to the input $\boldsymbol{x}$ is

$$\varphi_\beta(\boldsymbol{s}(\boldsymbol{x}) \cdot \boldsymbol{w}). \tag{4}$$

If $s_1(\boldsymbol{x}), \dots, s_M(\boldsymbol{x})$ are of the form $\varphi_{\beta_1}(\boldsymbol{x} \cdot \boldsymbol{v}_1), \dots, \varphi_{\beta_M}(\boldsymbol{x} \cdot \boldsymbol{v}_M)$ for some activating functions $\varphi_{\beta_1}, \dots, \varphi_{\beta_M}$ and weight vectors $\boldsymbol{v}_1, \dots, \boldsymbol{v}_M$, i.e. if the hidden units are neurons, then we call $\langle M, \boldsymbol{s}, \boldsymbol{w}\rangle$ a *perceptron* with one hidden layer. For simplicity we do not consider thresholds – they can be substituted by additional constant inputs.

A neural network (perceptron) with two hidden layers is a quintuple $\langle J, M, \boldsymbol{\sigma}, \boldsymbol{s}, \boldsymbol{w}\rangle$ where $\langle M, \boldsymbol{s}, \boldsymbol{w}\rangle$ is an output network with one hidden layer at the input of which is the response $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_J)$ of J units of the previous layer to the input $\boldsymbol{x}$ itself. Thus the response of whole network to an input $\boldsymbol{x}$ is

$$\varphi(\boldsymbol{s}(\boldsymbol{\sigma}(\boldsymbol{x})) \cdot \boldsymbol{w})$$

(in the case of perceptron all units are neurons). The definition of a network or perceptron with an arbitrary number of hidden layers follows from here.

The output neuron weights of an arbitrary network are free parameters the values of which are assumed to be specified by a learning procedure. Other similar parameters may be "hidden" in the units of the hidden layers. If the hidden unit is a neuron with input synaptic weights $\boldsymbol{w} \in R^i$ then it contains i parameters which are to be specified by the learning. If there are no free parameters in the hidden layers then one can use the learning rules for simple perceptrons described e. g. in Sec. 5.2 of Müller et al [12]. Otherwise one has to use the learning by error back-propagation (cf. Sec. 6.2 ibid).

We shall combine Assertion 1 with another well known fact established by Funahashi [5] and Hornik [8].

Assertion 2. For every closed and bounded subset $S \subset R^n$, and every continuous function $\delta : S \to R$ and positive ε, there exists a perceptron $\langle M, \boldsymbol{s}, \boldsymbol{w}\rangle$ with input $\boldsymbol{x} \in R^n$ and the linear response

$$\rho(\boldsymbol{x}) = \boldsymbol{s}(\boldsymbol{x}) \cdot \boldsymbol{w} \tag{5}$$

such that

$$\sup_{\boldsymbol{x} \in S} |\delta(\boldsymbol{x}) - \rho(\boldsymbol{x})| \leq \varepsilon. \tag{6}$$

Remark 1. The hidden units of the perceptron considered in Assertion 2 are neurons with input synaptic weights $\boldsymbol{v}_i \in R^n$ for $i = 1, \dots, k$ and appropriate parameters β_i in the sigmoidal activation functions.

Remark 2. As shown by Lapedes and Farber [10] (cf. also Sec. 6.4 of [12]), every "reasonable" function $\delta : R^k \to R$ can be approximated by a perceptron with at most two hidden layers of neurons (one layer if $k = 1$ and two layers otherwise). The "reasonable" means that δ can be approximated e. g. by piecewise linear functions, or by basis-spline functions widely used in numerical analysis. An advantage of the method of [10, 12] is that it is constructive, while the method of the authors of Assertion 2 guarantees the existence but says a little about the construction of desired perceptron. Sec. 25 in [12] describes a computer program PERFUNC for approximation of functions by the method of [10, 12].

Remark 3. In practical applications one usually encounters observations $\boldsymbol{x} = (x_1, \dots, x_n)$ with large sample sizes n. As follows from the iterative learning rules described in Sections 5, 6 of [12], one cannot expect reasonably precise specification of free perceptron parameters in a “reasonable time” if the number of these parameters grows with n. We speak about the learning in a “reasonable time” if the number of free perceptron parameters remains fixed for n increasing.

By combining Assertion 1 with Assertion 2 or Remark 2 we obtain approximations of Bayes decision rules $\delta^* : R^n \to A$ by means of perceptrons $\langle M, \boldsymbol{s}, \boldsymbol{w} \rangle$ or $\langle J, M, \boldsymbol{r}, \boldsymbol{s}, \boldsymbol{w} \rangle$ with inputs $\boldsymbol{y} \in R^k$ from the target spaces of sufficient statistics $\boldsymbol{\phi}(\boldsymbol{x})$. If Assertion 2 or Remark 2 with $k = 1$ are applicable then a perceptron $\langle M, \boldsymbol{s}, \boldsymbol{w} \rangle$ with one hidden layer of neurons is sufficient. Otherwise one has to use two hidden layers. The unknown synaptic weights of these perceptrons can be learned in a “reasonable time” if the dimension of $\boldsymbol{\phi}$ is not increasing with n.

Remark 4. The perceptrons $\langle M, \boldsymbol{s}, \boldsymbol{w} \rangle$ and $\langle J, M, \boldsymbol{\sigma}, \boldsymbol{s}, \boldsymbol{w} \rangle$ with inputs $\boldsymbol{y} \in R^k$ approximate the function $\delta(\boldsymbol{y})$ figuring in (1). Approximations of the Bayes rule $\delta^*(\boldsymbol{x})$ are obtained by feeding these perceptrons with inputs $\boldsymbol{\phi}(\boldsymbol{x})$. This in fact leads to new perceptrons with inputs $\boldsymbol{x} \in R^n$ and two or three hidden layers respectively, where the first hidden layer consists of k units with responses $\phi_1(\boldsymbol{x}), \ldots, \phi_k(\boldsymbol{x})$. Since hidden layer contains no free parameters, learning of the new and original perceptrons coincide.

Example 1. Let the dimension of $\boldsymbol{\phi}$ be $k = 1$ and let us consider a function $\delta(y)$ of variable $y \in R$. Then the perceptron $\langle M, \boldsymbol{s}, \boldsymbol{w} \rangle$ with one hidden layer of neurons and input y for approximation of the function $\delta(y)$ has the form presented in Figure 1. The activation functions $f_j(h)$ in the hidden layer are arbitrary, e. g. they may be identical mappings $f_j(h) = h$ or they may belong to the family of sigmoidal functions considered in (3) for parameters β_j from the extended real line $\overline{R} = [-\infty, \infty]$.

Fig. 1. Perceptron $\langle M, \boldsymbol{s}, \boldsymbol{w} \rangle$ with one hidden layer of neurons.

Example 2. Let the components x_i of $\boldsymbol{x}$ be independently distributed by the Bernoulli P_θ with $P_\theta(1) = 1 - P_\theta(0) = \theta$ for $\theta \in (0, 1) = \Theta$. It is known that

$$\phi(\boldsymbol{x}) = \sum_{i=1}^{n} x_i$$

is a binomially distributed sufficient statistics for the family $\{P_\theta^n : \theta \in (0, 1)\}$. If we consider on $(0, 1)$ the beta prior density

$$\pi(\theta) = \frac{\theta^{a-1}\,(1-\theta)^{b-1}}{B(a,b)} \quad \text{for } a,\, b > 0,$$

and if we put $A = (0,1)$ and consider the squared-error loss function $L(\theta, a) = (\theta - a)^2$, then the Bayes rule is (see Berger [1] or p. 351 in Mood et al [11])

$$\delta^*(\boldsymbol{x}) = \frac{\sum_1^n x_i + a}{n+a+b} = \delta(\phi(\boldsymbol{x})) \quad \text{for} \quad \delta(y) = \frac{y+a}{n+a+b}.$$

Thus the neuron with input $(y, 1)$, synaptic weights

$$(v_1, v_2) = \left(\frac{1}{n+a+b}, \frac{a}{n+a+b}\right)$$

and linear response

$$s(y,1) = (v_1, v_2) \cdot (y, 1)$$

exactly imitates the Bayes version δ (this neuron is a special case of perceptron of Figure 1 with $M = 1$, $f_1(h) = h$ and $w_1 = 1$). The Bayes rule $\delta^*(\boldsymbol{x})$ is realized by the perceptron $\langle 2, 1, (\sigma_1, \sigma_2), s, 1\rangle$ with two hidden layers, hidden responses

$$\sigma_1(\boldsymbol{x}) = \phi(\boldsymbol{x}) = \sum_{i=1}^n x_i, \quad \sigma_2(\boldsymbol{x}) = 1, \quad \text{and} \quad s(\sigma_1, \sigma_2) = (v_1, v_2) \cdot (\sigma_1, \sigma_2),$$

and the output neuron response $\rho(s) = 1 \cdot s = s$. This is the extension of the original perceptron $\langle 1, s, 1\rangle$ considered in Remark 4. Its scheme is presented in Figure 2. Note that both perceptrons under consideration are capable of adaptation to arbitrary parameters a, b of the prior distribution for all possible sample sizes n.

Fig. 2. Perceptron realizing the Bayes rule δ^*.

Remark 5. Learning of the perceptrons $\langle M, \boldsymbol{s}, \boldsymbol{w}\rangle$ and $\langle J, M, \boldsymbol{\sigma}, \boldsymbol{s}, \boldsymbol{w}\rangle$ consists in the presentation of pairs $(\boldsymbol{y}_1,\, \delta(\boldsymbol{y}_1)), \dots, (\boldsymbol{y}_N,\, \delta(\boldsymbol{y}_N))$ for $\boldsymbol{y}_i = \boldsymbol{\phi}(\boldsymbol{x}_i)$ corresponding to the observed data vectors $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$. The values $\delta(\boldsymbol{y})$ are arguments of minima of integrals

$$I(a) = \int_{R^m} L(\boldsymbol{\theta}, a)\, g_{\boldsymbol{\theta}}(\boldsymbol{y})\, \pi(\boldsymbol{\theta})\, \mathrm{d}\nu(\boldsymbol{\theta})$$

on the action space A (cf. (2) and proof of Assertion 1). If these arguments can be evaluated explicitly as functions of $\boldsymbol{y}$ on the whole domain R^k, as it is in Example 2, then the practical advantage of the perceptron realization of δ or δ^* is limited. E. g. in Example 2 this advantage is limited to the adaptivity of the resulting perceptron to the prior distribution parameters a, b which may be apriori unknown. But these nonlinear regression parameters can be evaluated from the empirical evidence

$(y_1,\, \delta(y_1)), \ldots, (y_N,\, \delta(y_N))$ directly, without the perceptron. Moreover, the direct statistical method is at least as efficient as the perceptron method. Thus the only argument which remains in this case in favour of the perceptron is that it represents a relatively simple automaton capable of adaptation in a nontrivial statistical environment. If however the explicit form of $\delta(\boldsymbol{y})$ is unknown and the evaluation of $\arg\min I(a)$ is a difficult task, then there is a stronger argument in favour of the perceptrons under consideration. Namely, by being "learned", these perceptrons extrapolate the knowledge concentrated in the ensemble $(\boldsymbol{y}_1,\, \delta(\boldsymbol{y}_1)), \ldots, (\boldsymbol{y}_N,\, \delta(\boldsymbol{y}_N))$ on the whole domain of $\boldsymbol{y}$ by providing approximations $\rho(\boldsymbol{y})$ and $\rho(\boldsymbol{\phi}(\boldsymbol{x}))$ to $\delta(\boldsymbol{y})$ and $\delta^*(\boldsymbol{x})$ at all remaining points $\boldsymbol{y} \in R^k - \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$. In similar situations perceptron applications proved to be useful in the past (cf. e.g. [12, 13]).

3. BAYES DISCRIMINATION

In this section we consider a special variant of the statistical decision model of previous section, with the prior distribution concentrated on just two points $\boldsymbol{\theta}_1,\, \boldsymbol{\theta}_2 \in \Theta$. By $\pi_1,\, \pi_2$ we denote prior probabilities of these points and we shall assume that π_1 and π_2 are positive with $\pi_1 + \pi_2 = 1$. The set of actions A consists of integers 1 and 2 and the loss $L(\boldsymbol{\theta}_i, j)$ is assumed to be zero if $i = j$ and positive if $i \neq j$. This is the model of discrimination between observations $\boldsymbol{x}$ generated by the law $p_{\boldsymbol{\theta}_1}$ and those generated by $p_{\boldsymbol{\theta}_2}$ (for more details we refer to Hand [7]).

Denote by

$$\lambda_1 = L(\boldsymbol{\theta}_1, 2) \quad \text{and} \quad \lambda_2 = L(\boldsymbol{\theta}_2, 1)$$

the losses of misdiscrimination. Then it follows from (2) that the Bayes discrimination $\delta^* : R^n \to \{1, 2\}$ is defined by the condition

$$\delta^*(\boldsymbol{x}) = \arg\max \phi(i) \quad \text{for} \quad \phi(i) = \lambda_i\, \pi_i\, p_{\boldsymbol{\theta}_i}(\boldsymbol{x}), \quad i \in \{1, 2\}.$$

It follows from here that the well known rule

$$\delta^*(\boldsymbol{x}) = \left\{ \begin{array}{c} 1 \\ 2 \end{array} \right. \quad \text{if} \quad \lambda_1\, \pi_1\, p_{\boldsymbol{\theta}_1}(\boldsymbol{x}) \quad \left\{ \begin{array}{ll} \geq & \lambda_2\, \pi_2\, p_{\boldsymbol{\theta}_2}(\boldsymbol{x}) \\ < & \lambda_2\, \pi_2\, p_{\boldsymbol{\theta}_2}(\boldsymbol{x}) \end{array} \right. \tag{7}$$

represents the Bayes discrimination.

Let us suppose that the observations are exponentially distributed. This means that the parameter space Θ is an open convex subset of R^m, and that there exists a mapping $\boldsymbol{T} : R^n \to R^m$ such that

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \exp(\boldsymbol{\theta} \cdot \boldsymbol{T}(\boldsymbol{x}) - \psi(\boldsymbol{\theta})) \quad \text{for all} \quad \boldsymbol{\theta} \in \Theta,\ \boldsymbol{x} \in R^n, \tag{8}$$

where ·

$$\psi(\boldsymbol{\theta}) = \ln \int \exp(\boldsymbol{\theta} \cdot \boldsymbol{T}(\boldsymbol{x}))\, \mathrm{d}\mu(\boldsymbol{x}).$$

Let us assume that the family (8) is not overparametrized, i.e. that $(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2) \cdot \boldsymbol{T}(\boldsymbol{x})$ is not μ-almost everywhere constant. This means that distributions (8) are for different parameters $\boldsymbol{\theta}$ different (the family is identifiable by the parameter $\boldsymbol{\theta}$).

Note that (8) is a standard form of exponential distributions. The more familiar form

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = a(\boldsymbol{\theta})\, b(\boldsymbol{x}) \exp(\boldsymbol{c}(\boldsymbol{\theta}) \cdot \boldsymbol{T}(\boldsymbol{x}))$$

can be transformed into the standard form by the substitution $\boldsymbol{c}(\boldsymbol{\theta}) \to \boldsymbol{\theta}$ and by a modification of μ (cf. [2]).

Now we can formulate the main result of this section.

Assertion 3. If the data are exponentially distributed then the Bayes discrimination (7) coincides with the response

$$\rho(\boldsymbol{x}) = 1 + 1_{(-\infty,0)}(\boldsymbol{s}(\boldsymbol{x}) \cdot \boldsymbol{w}) \quad \text{(cf. (3))}$$

of the perceptron $\langle m+2, \boldsymbol{s}, \boldsymbol{w} \rangle$ defined by

$$\boldsymbol{s}(\boldsymbol{x}) = (\boldsymbol{T}(\boldsymbol{x}), 1, 1) \in R^{m+2} \qquad \text{(cf. (9))}$$

and

$$\boldsymbol{w} = (\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2,\, \psi(\boldsymbol{\theta}_2) - \psi(\boldsymbol{\theta}_1),\, \ln(\lambda_1 \pi_1) - \ln(\lambda_2 \pi_2)) \in R^{m+2}.$$

Proof. By (7), $\delta^*(\boldsymbol{x}) = 1$ if and only if

$$\ln \frac{p_{\boldsymbol{\theta}_1}(\boldsymbol{x})}{p_{\boldsymbol{\theta}_2}(\boldsymbol{x})} + \ln \frac{\lambda_1 \pi_1}{\lambda_2 \pi_2} \geq 0. \tag{9}$$

But according to (8)

$$\ln \frac{p_{\boldsymbol{\theta}_1}(\boldsymbol{x})}{p_{\boldsymbol{\theta}_2}(\boldsymbol{x})} = (\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2) \cdot T(\boldsymbol{x}) + \psi(\boldsymbol{\theta}_2) - \psi(\boldsymbol{\theta}_1).$$

Thus (9) holds if and only if $\boldsymbol{s}(\boldsymbol{x})$ and $\boldsymbol{w}$ considered in Assertion 3 satisfy the relation $\boldsymbol{s}(\boldsymbol{x}) \cdot \boldsymbol{w} \geq 0$, i. e. if and only if $\rho(\boldsymbol{x}) = 1$. □

The perceptron of Assertion 3 is the extension considered in Remark 4 of the simple perceptron of Figure 3. Inputs $y_1, \ldots, y_m$ of this perceptron are components $T_1(\boldsymbol{x}), \ldots, T_m(\boldsymbol{x})$ of the statistic $\boldsymbol{T}(\boldsymbol{x})$ which is sufficient for the exponential family (8).

Fig. 3. Output neuron of $\langle m+2, \boldsymbol{s}, \boldsymbol{w} \rangle$ in Assertion 3.

The statistic $\boldsymbol{T}$ describes the response of hidden layer of the perceptron considered in Assertion 3. If this statistics is known then the hidden layer contains no free parameters requiring to be learned. Such parameters may contain only the output neuron. If the parameters $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ of distributions governing the discriminated observations are known, and also the losses λ_1, λ_2 and prior probabilities π_1, π_2 are known, then nothing remains to be learned at all. Otherwise some or all synaptic weights $w_1, \ldots, w_{m+2}$ of the output neuron are to be specified by learning. But this learning is much easier than that the learning considered in previous section. This is due to the fact the perceptron here is simple, i. e. contains no hidden units.

If the statistic $\boldsymbol{T}$ is unknown and only the dimension m of the exponential model is given, then one can approximate the components $T_i(\boldsymbol{x})$ of $\boldsymbol{T}(\boldsymbol{x})$ by responses $\rho_i(\boldsymbol{x})$ of perceptrons considered in Assertion 2 or Remark 2. This leads to the approximation of the scalar product $\boldsymbol{s}(\boldsymbol{x}) \cdot \boldsymbol{w}$ (and, consequently, of the Bayes discrimination $\delta^*(\boldsymbol{x})$) by the perceptron of Figure 4. In this figure the boxes with $\rho_1(\boldsymbol{x}), \ldots, \rho_m(\boldsymbol{x})$ are perceptrons of the type considered in Assertion 2 or Remark 2. Therefore the perceptron of Figure 3 has two or three hidden layers. Learning in such perceptrons is rather slow but possible (see [12]).

Fig. 4. Perceptron approximation to $\boldsymbol{s}(\boldsymbol{x}) \cdot \boldsymbol{w}$.

Remark 6. In most applications the losses and prior probabilities are considered symmetric, i. e. $\lambda_1 = \lambda_2$ and $\pi_1 = \pi_2$. In such situations the dimension of above considered perceptrons can be reduced by one, i. e. it suffices to consider $\langle m{+}1, \boldsymbol{s}, \boldsymbol{w}\rangle$ with $\boldsymbol{s}(\boldsymbol{x}) = (\boldsymbol{T}(\boldsymbol{x}),\, 1) \in R^{m+1}$ and $\boldsymbol{w} = (\boldsymbol{\theta}_1 - \boldsymbol{\theta}_2,\, \psi(\boldsymbol{\theta}_2) - \psi(\boldsymbol{\theta}_1)) \in R^{m+1}$.

4. EXAMPLE: CLASSIFICATION OF NORMAL DATA

Let us consider the exponential family (8) with $n = 1$, $m = 2$ and with the Lebesgue measure μ on R. Such a family is specified by two statistics $T_1(x)$ and $T_2(x)$. Then for $\boldsymbol{\theta} = (\alpha, \beta)$

$$\psi(\alpha, \beta) = \ln \int \exp(\alpha\, T_1(x) + \beta\, T_2(x))\, \mathrm{d}x.$$

Our attention will be restricted to the symmetric case $\lambda_1 = \lambda_2$ and $\pi_1 = \pi_2$.

Consider the particular functions $T_1(x) = x$ and $T_2(x) = -x^2$. Then $\Theta = R \times (0, \infty)$ and

$$\psi(\alpha, \beta) = \frac{1}{2}\left[\frac{\alpha^2}{2\beta} + \ln\frac{\pi}{\beta}\right]$$

for all $(\alpha, \beta) \in R \times (0, \infty)$. It is easy to verify that then (8) is the normal family with mean and variance

$$\mu = \frac{\alpha}{2\beta} \quad \text{and} \quad \sigma^2 = \frac{1}{2\beta}.$$

Indeed,

$$p_{\alpha,\beta}(x) = e^{\alpha\, T_1(x) + \beta\, T_2(x) - \psi(\alpha,\beta)} = \frac{e^{\alpha x - \beta x^2 - \alpha^2/4\beta}}{\sqrt{\frac{\pi}{\beta}}} = \frac{e^{-(x-\mu)^2/2\sigma^2}}{\sqrt{2\pi\sigma^2}}.$$

In the present model, taking into account Remark 6, the perceptron of Figure 4 reduces to that of Figure 5 where the boxes with $\rho_1(x)$ and $\rho_2(x)$ represent perceptrons with the hidden layer of neurons considered in Remark 2. Thus the perceptron of Figure 5 contains two hidden layers of neurons. This perceptron approximates the Bayes discrimination $\delta^*(x)$ which is for

Fig. 5. Perceptron approximation of Bayes discrimination $\delta^*(x)$. The response is $1 + 1_{(-\infty,0)}(\rho(x))$.

$$\boldsymbol{\theta}_1 = (\alpha_1, \beta_1) \quad \text{and} \quad \boldsymbol{\theta}_2 = (\alpha_2, \beta_2)$$

defined in accordance with Assertion 3 by

$$\delta^*(x) = 1 + 1_{(-\infty,0)}(T_1(x)\, w_1 + T_2(x)\, w_2 + w_3), \tag{10}$$

for

$$w_1 = (\alpha_1 - \alpha_2), \quad w_2 = (\beta_1 - \beta_2), \quad w_3 = \psi(\alpha_2, \beta_2) - \psi(\alpha_1, \beta_1). \tag{11}$$

Let x be the above considered normal data for

$$\boldsymbol{\theta}_1 = (0,\, (2\sigma_1)^{-1}) \quad \text{and} \quad \boldsymbol{\theta}_2 = (0,\, (2\sigma_2)^{-1}) \quad \text{where } 0 < \sigma_1 < \sigma_2$$

i.e. for $(\mu_1, \sigma_1^2) = (0, \sigma_1^2)$ and $(\mu_2, \sigma_2^2) = (0, \sigma_2^2)$. It follows from (10) and (11) that the Bayes discrimination is

$$\delta^*(x) = 1 + 1_{(-\sigma_0, \sigma_0)}(x), \tag{12}$$

where $\sigma_0 > 0$ is solution of the equation $p_{\theta_1}(x) = p_{\theta_2}(x)$, i.e.

$$\sigma_0 = \left(\frac{\sigma_1^2 \sigma_2^2 \ln(\sigma_2/\sigma_1)^2}{\sigma_2^2 - \sigma_1^2} \right)^{1/2}.$$

This discrimination function is approximated with the help of perceptron approximations $\rho_1(x)$ and $\rho_2(x)$ to statistics $T_1(x) = x$ and $T_2(x) = -x^2$ considered in Figure 5. According to Remark 2, we used two-layer perceptrons with 3 neurons in each layer and with the sigmoidal activation functions with $\beta = 0.7$ in these neurons and linear activation in the output neuron. Learning has been performed by using data $(x_1,\, \delta^*(x_1)), \dots, (x_N,\, \delta^*(x_N))$ where $x_1, \dots, x_N$ are independent realizations of random variable with the mixed density

$$p(x) \overset{\triangle}{=} \frac{1}{2}\left(p_{\boldsymbol{\theta}_1}(x) + p_{\boldsymbol{\theta}_2}(x)\right) = \frac{1}{2\sqrt{2\pi}}\left(\frac{e^{-x^2/2\sigma_1^2}}{\sigma_1} + \frac{e^{-x^2/2\sigma_2^2}}{\sigma_2}\right) \tag{13}$$

and $N = 2000$. Data were simulated by using a pseudorandom generator. Distribution of a standard normal output from this generator is presented in Figure 6. Learning of weights of all 7 neurons in the network of Figure 5 was carried out by the standard error back-propagation algorithm described e. g. in Müller et al [12] with the constant learning rate $\varepsilon = 0.007$.

Fig. 6. Histogram of 10^5 standard normal data with step 10^2, and the standard normal density.

The experiments were done out for $\sigma_1 = 1$ and 14 different values of σ_2, with the theoretical Bayes error P_E^{Bayes} varying between 0.5 and 0. For each pair (σ_1, σ_2) we carried out 20 experiments with randomly selected initial weights. At the end of each experiment the learned network classified 98 000 data randomly selected according to density (12) and the error frequency P_E^{perc} was calculated. In all experiments this error was reasonably close to the smallest theoretically achievable Bayes error

$$\begin{aligned} P_E^{\text{Bayes}} &= \frac{1}{2}\left[1-(F(\sigma_0)-F(-\sigma_0))+F(\sigma_0/\sigma_2)-F(\sigma_0/\sigma_2)\right] \\ &= \frac{1}{2}-F(\sigma_0)+F(\sigma_0/\sigma_2), \end{aligned}$$

where $F(x)$ denotes the standard normal distribution function. The best 14 of the 20 experiments are presented in Tab. 1. We see that the quality of classifiers obtained without using the knowledge of statistical models is excellent.

Table 1. Performances of the best of 20 realized experiments.

σ_1	σ_2	σ_0	P_E^{Bayes}	P_E^{perc}	$P_E^{\text{perc}} - P_E^{\text{Bayes}}$
1	1.5	1.21	0.4012	**0.4024**	0.0012
1	2.0	1.36	0.3355	**0.3358**	0.0003
1	2.5	1.47	0.2898	**0.2913**	0.0004
1	3.0	1.57	0.2567	**0.2560**	0.0003
1	3.5	1.65	0.2303	**0.2293**	0.0010
1	4.0	1.72	0.2091	**0.2086**	0.0005
1	4.5	1.78	0.1912	**0.1907**	0.0005
1	5.0	1.83	0.1765	**0.1756**	0.0009
1	5.5	1.88	0.1637	**0.1626**	0.0011
1	6.0	1.92	0.1529	**0.1516**	0.0013
1	6.5	1.96	0.1437	**0.1425**	0.0012
1	7.0	1.99	0.1354	**0.1342**	0.0012
1	7.5	2.02	0.1282	**0.1270**	0.0012
1	8.0	2.05	0.1216	**0.1202**	0.0014

We were also interested in whether, or to what extent, the responses $\rho_1(x)$ and $\rho_2(x)$ of the two-layered subnetworks approximate the desired functions $T_1(x) = x$ and $T_2(x) = x^2$. Of course, this approximation is irrelevant (and cannot be achieved by any empirical means) outside the effective domain of the distribution (13), where no or few data are realized. Also, since the network is learned as a whole, the approximation must be modulo linear transforms, i. e. arbitrary $a_i\, T_i(x) + b_i$ may be realized. We verified that this evidently took place in great majority of experiments. Figures 7 – 10 present typical examples, two experiments for $\sigma_2 = 2.5$ and two for $\sigma_2 = 4$. Together with the overall network response $\rho(x)$ and the responses $\rho_1(x)$ and $\rho_2(x)$ are presented best linear and quadratic approximations, achieving

$$\min_{a_i,b_i\in R} \int_{-1}^{1} (a_i x + b_i - \rho_1(x))^2\, p(x)\mathrm{d}x$$

and $$\min_{a_i,b_i,c_i\in R} \int_{-1}^{1} (a_i + b_i x + c_i x^2 - \rho_2(x))^2\, p(x)\mathrm{d}x$$

for $p(x)$ defined by (13). The subdomain $(-1, 1)$ is slightly overemphasized in these calculations but, nevertheless, we see that one of the functions $\rho_1(x)$, $\rho_2(x)$ in the effective domain is always closer to the quadratic and the other to the linear function. Thus the network really "learned" the sufficient statistics.

Fig. 7. The overall network response $\rho(x)$ (dotted line), the responses $\rho_1(x)$, $\rho_2(x)$ of two-layer perceptrons and their L_2-norm projections on linear and quadratic functions. The case $\sigma_1 = 1$ and $\sigma_2 = 2.5$.

Fig. 8. As in Figure 7, but a different experiment.

Fig. 9. The overall network response $\rho(x)$ (dotted line), the responses $\rho_1(x)$, $\rho_2(x)$ of two-layer perceptrons and their L_2-norm projections on linear and quadratic functions. The case $\sigma_1 = 1$ and $\sigma_2 = 4$.

Fig. 10. As in Figure 9, but a different experiment.

5. EXAMPLE: CLASSIFICATION OF GEOMETRIC BROWNIAN MOTIONS

Let $W = (W_t : t \geq 0)$ be a Wiener process. Then a process $X = (X_t : t \geq 0)$ satisfying the stochastic differential equation

$$d\,X_t = \theta\, X_t\, \mathrm{d}t + X_t\, \mathrm{d}W_t, \quad t > 0,$$

with $X_0 = 1$ is called a *geometric Brownian motion*. In spite of that X does not have independent increments, the likelihood of the trajectory $(X_t : 0 \leq t \leq T)$ is a function of the final state X_T only,

$$L_{[0,T]}(\theta) = \exp\left\{\theta \log X_T + \frac{\theta^2}{2}\, T\right\}, \quad \theta \in R.$$

The process is thus exponentially distributed and the final state X_T is a sufficient statistics.

We shall consider two types of trajectories corresponding to $\theta = \theta_0$ and $\theta = -\theta_0$, and we shall put

$$\Lambda = \theta_0\, T.$$

Using the results of Section 3 we see that the Bayes classifier is of the form

$$\delta_*(X_t : 0 \leq t \leq T) = 1_{(0,\infty)}(w_1\, X_T + w_2)$$

for appropriate weights w_1 and w_2. We experimented with the values $\Lambda = 0.1$, $\Lambda = 0.5$ and $\Lambda = 1$. The initial perceptron weights were $(w_{10}, w_{20}) = (0, 0)$, we used the same source of random data as in Section 4, and we applied the classical perceptron learning rule (cf. Müller et al [12]) with the variable learning rate $\varepsilon(n) = (0.05)^n$. We checked after various numbers n of learning steps on 1 000 new samples the frequency of error $P_E^{\text{perc}}(n)$.

We see from Figure 11 and Figure 12 that approximately 100 learning steps are sufficient to stabilize P_E^{perc} for any $0,1 \leq \Lambda < 1$. Similarly as in the previous section, it is easy to verify that P_E^{perc} is stabilized in the neighborhood of P_E^{Bayes}.

6. BAYES CLASSIFICATION

The general decision problem of Section 2 reduces to the classification problem by taking the prior distribution concentrated on a finite subset $\{\theta_1, \dots, \theta_r\} \subset \Theta$ and by putting $A = \{1, \dots, r\}$ (cf. Hand [7]). The nonzero losses are

$$\lambda_{ij} = L(\theta_i, \theta_j) \quad \text{for } i \neq j.$$

We restrict ourselves to the most important case where all these losses coincide.

If $\pi_1, \dots, \pi_r$ are prior probabilities of $\theta_1, \dots, \theta_r$ then it follows from (2) that the Bayes classification $\delta^* : R^n \to \{1, \dots, m\}$ is defined by the condition

$$\delta^*(\boldsymbol{x}) = \arg\max \phi(i) \quad \text{for} \quad \phi(i) = \pi_i\, p_{\boldsymbol{\theta}_i}(\boldsymbol{x}),\ i \in \{1, \dots, r\}.$$

Fig. 11. $P_E^{\mathrm{perc}}(n)$ versus number of learning steps n for $\Lambda = 0.1$ (upper line) and $\Lambda = 0.5$ (lower line).

Fig. 12. $P_E^{\mathrm{perc}}(n)$ versus number of learning steps for $\Lambda = 1$.

Let us consider for $i \neq j$ the Bayes discriminations $\delta^*_{ij}(\boldsymbol{x})$ defined by (7) in the discrimination problems with the prior probabilities $\pi_i/(\pi_i + \pi_j)$ and $\pi_j/(\pi_i + \pi_j)$

and distributions $P_{\boldsymbol{\theta}_i}$ and $P_{\boldsymbol{\theta}_j}$. Put

$$\delta^*(\boldsymbol{x}) = i \quad \text{if} \quad \delta^*_{ij}(\boldsymbol{x}) = i \quad \text{for all } j \in \{1, \dots, r\}.$$

It follows from (7) that this defines a mapping $\delta^* : R^n \to \{1, \dots, r\}$ which is a Bayes classification in the classification problem under consideration. Thus we have proved

Assertion 4. The Bayes classification $\delta^*(\boldsymbol{x})$ under consideration can be evaluated by means of the Bayes discriminations $\delta^*_{ij}(x)$ considered above.

Assertion 4 implies that the perceptron realizations of Bayes classifications, or perceptron approximations to these classifications, can be obtained from the perceptron realizations or approximations considered in Section 3. Therefore we do not go into details.

7. CONCLUSIONS

In statistical models with known sufficient statistics of dimension not increasing with the sample size, we have found the possibility to approximate Bayes decision functions by a perceptron with at most three hidden layers, of complexity not increasing with the sample size. The need of learning is restricted to synaptic weights of neurons from two hidden layers plus one output neuron.

In the particular discrimination problem with exponential distributions it is shown that the number of hidden layers can be reduced to one and the need of learning is restricted just to synaptic weights of the output neuron.

If we know only the dimension of exponential distributions, then the Bayes classification can be approximated by a perceptron with at most three hidden layers. The conclusions of the first paragraph about complexity and the necessity of learning remain valid for this perceptron.

The results concerning discrimination can be extended to Bayes classification.

A frequent objection against neural network solutions of statistical problems is that the statistical solutions are usually more efficient (see Ripley [13]). The efficiency argument is true, but the loss of efficiency is at least to some extent compensated by the algorithmic simplicity. Moreover, the applicability of neural nets beyond the scope of models satisfying the mathematical assumptions of statistical algorithms is demonstrated by a considerable neural network literature. The fact that the most efficient methods based on likelihood ratio (like e. g. the Bayes discrimination and classification considered in this paper) can be misleading if their assumptions are not strictly fulfilled has been proved by theoretical results and simulations in robust statistics (see e. g. Hampel et al [6]).

(Received January 8, 1997.)

REFERENCES

[1] J. O. Berger: Statistical Decision Theory and Bayesian Analysis. Second edition. Springer, New York 1985.

[2] L. D. Brown: Fundamentals of Statistical Exponential Families. Lecture Notes 9. Inst. of Mathem. Statist., Hayward, California 1986.

[3] H. H. Bock: A clustering technique for maximizing ϕ-divergence, noncentrality and discriminating power. In: Analyzing and Modelling Data and Knowledge (M. Schader, ed.), Springer, Berlin 1992, pp. 19–36.

[4] P. Devijver and J. Kittler: Pattern Recognition: A Statistical Approach. Prentice Hall, Englewood Cliffs 1982.

[5] K. Funahashi: On the approximate realization of continuous mappings by neural networks. Neural Networks *2* (1989), 183–192.

[6] F. R. Hampel, P. J. Rousseeuw, E. M. Ronchetti and W. A. Stahel: Robust Statistics: The Approach Based on Influence Functions. Wiley, New York 1986.

[7] D. J. Hand: Discrimination and Classification. Wiley, New York 1981.

[8] K. Hornik, M. Stinchcombe and H. White: Multilayer feedforward networks and universal approximation. Neural Networks *2* (1989), 359–366.

[9] U. Küchler and M. Sørensen: Exponential families of stochastic processes: A unifying semimartingale approach. Internat. Statist. Rev. *57* (1989), 123–144.

[10] A. S. Lapedes and R. H. Farber: How neural networks work. In: Evolution, Learning and Cognition (Y. S. Lee, ed.), World Scientific, Singapore 1988, pp. 331–340.

[11] A. M. Mood, F. A. Graybill and D. C. Boes: Introduction to the Theory of Statistics. Third edition. McGraw–Hill, New York 1974.

[12] B. Müller, J. Reinhard and M. T. Strickland: Neural Networks. Second edition. Springer, Berlin 1995.

[13] B. D. Ripley: Statistical aspects of neural networks. In: Networks and Chaos (O. E. Barndorff–Nielsen, J. L. Jensen and W. S. Kendall, eds.), Chapman and Hall, London 1993. pp. 40–123.

[14] I. Vajda: About perceptron realizations of Bayesian decisions about random processes. In: IEEE International Conference on Neural Networks, vol. 1, IEEE, 1996, pp. 253–257.

Ing. Igor Vajda, DrSc., 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: vajda@utia.cas.cz

Ing. Belomír Lonek, Faculty of Nuclear and Physical Engineering, Czech Technical University, Trojanova 13, 120 00 Praha 2. Czech Republic.
e-mail: lonek@km1.fjfi.cvut.cz

Ing. Viktor Nikolov, SEFIRA s. r. o., Káranská 41, 108 00 Praha 10.
e-mail: sefira@vol.cz.

Ing. Arnošt Veselý, CSc., Czech Agricultural University, Kamýcká 129, 160 00 Praha 6. Czech Republic.
e-mail: vesely@pef.vsz.cz