entropy

MDPI

Article

Entropy "2"-Soft Classification of Objects

Yuri S. Popkov [1,2,*], Zeev Volkovich [3], Yuri A. Dubnov [1,2], Renata Avros [3] and Elena Ravve [3]

1 Institute for Systems Analysis of Federal Research Center "Computer Science and Control", Moscow 117312, Russia; yury.dubnov@phystech.edu
2 Intelligent Technologies in System Analysis and Management, National Research University Higher School of Economics, Moscow 125319, Russia
3 Department of Software Engineering, ORT Braude College, Karmiel 2161002, Israel; vlvolkov@braude.ac.il (Z.V.); r_avros@braude.ac.il (R.A.); cselena@braude.ac.il (E.R.)
* Correspondence: popkov@isa.ru; Tel.: +7-499-135-42-22

Academic Editor: Dawn E. Holmes
Received: 10 March 2017; Accepted: 18 April 2017; Published: 20 April 2017

Abstract: A proposal for a new method of classification of objects of various nature, named "2"-soft classification, which allows for referring objects to one of two types with optimal entropy probability for available collection of learning data with consideration of additive errors therein. A decision rule of randomized parameters and probability density function (PDF) is formed, which is determined by the solution of the problem of the functional entropy linear programming. A procedure for "2"-soft classification is developed, consisting of the computer simulation of the randomized decision rule with optimal entropy PDF parameters. Examples are provided.

Keywords: randomization; entropy; learning collection; machine learning; objects classification; randomized machine learning

1. Introduction

The problem of object classification is highly relevant in contemporary theoretical and applied science. Objects, which are subject to classification, can be text documents, audio, video and graphic objects, events, etc. The *"m"-soft classification* means the attribution of the object to the appropriate *m* class with a certain probability, unlike the *"m"-hard classification*, when no alternative distribution of objects by classes is performed. Here, we consider "2"-soft classification, which is the basis for the classification by *m* classes.

"2"-soft classification is useful in many applied problems, with the exception, perhaps, of "ideal" ones, where all classes, object specification, and data are absolutely accurate. The fact is that real classification problems are immersed in significantly undefined environments. When it comes to data, they are received with errors, omissions, questionable reliability, and different timelines. The formation of decision rule models and their parameterization is not a formalized and subjective process, depending on the knowledge and experience of a researcher. By minimizing an empirical risk of decision rule model in the learning process, we get parameter evaluations for the existing amounts of data (precedents) and for the accepted parameterized model, i.e., evaluations are conditional. How will they, and consequently results of the classification, would behave with other precedents and with other parameterization model remains unclear. Methods of "2"-soft classification are directed to indicate a possible approach to overcome uncertainty factors. The idea is to make a decision rule randomized, and not arbitrarily randomized, but so that its entropy, as a measure of uncertainty, was at the maximum. It would allow for generating an ensemble of the best solutions at the highest uncertainty.

However, among soft and hard classification, fundamental differences exist: the structures of their procedures are similar and based on a more general concept of machine learning by precedents. A huge amount of work is devoted to this issue. Relevant references to them can be found in monographs [1–8], lectures [9,10] and reviews [11–13]. The recent fundamental works [6,14,15] clarify the vast diversity of classification algorithms and its learning procedures.

Within the general concept of machine learning, its modification had been proposed: *Randomized Machine Learning* (*RML*) [16]. An idea of the randomization is expanding to data and parameters of decision rules. This means that the model parameters of decision rules and data errors are assumed to be randomized in an appropriate way. The difference from existing machine learning procedures is that *RML* procedures are built not for optimal evaluations of the model parameters, but their probability density function (*PDF*) and evaluation of the "worst" data errors of *PDF*. In the *RML*, as a criterion of evaluation optimality, generalized information entropy is used, maximization of which was carried out on a set described by the system of empirical balances with collections of learning data.

The principle of maximum entropy has already been used in the domain of machine learning—for example, for speech recognition and text classification problems [17] and even for deep neural net parameters estimation [18]. The main advantage of this technique is its robustness to over-fitting in the presence of data errors within small data sets [19,20]. It is demonstrated by the classification experiments with the additive random noise presented in this paper.

2. Statement of the Problem

Suppose that there are two collections of objects: $\mathcal{E} = \{e_1, \ldots, e_h\}; \mathcal{T} = \{t_1, \ldots t_r\}$, which must be distributed between two classes. Objects in both collections are performed by vectors, whose components are variable features that characterize an object and measured quantitatively:

$$\mathcal{E} = \{\mathbf{e}^{(1)}, \ldots, \mathbf{e}^{(h)}\}, \qquad \mathcal{T} = \{\mathbf{t}^{(1)}, \ldots, \mathbf{t}^{(r)}\}. \tag{1}$$

We will assume that dimensions of the vectors in both collections are the same, i.e., $(\mathbf{e}^{(i)}, \mathbf{t}^{(j)}) \in R^n$. Collection $\mathcal{E}$ is used for learning, and collection $\mathcal{T}$ for its classification and testing.

The objects in both collections are marked by belonging to a corresponding class: if object e_l or t_k belongs to the first class, it will be assigned 1, or 0 if it belongs to the second one. Consequently, the learning collection is characterized by a vector of answers $\mathbf{y} = \{y_1, \ldots, y_h\}$ with components equal to 0 or 1, and a testing collection to vector of responses $\mathbf{z} = \{z_1, \ldots, z_r\}$ (at the end of last century it was known as learning with a teacher [2]). Numbers of these vector components correspond to object numbers in learning and testing collections.

2.1. Learning

Availability of learning collection allows for hypothesizing about the existence of function (decision rule) $F : \mathcal{E} \times \mathcal{S}_2 \to \mathbf{y}$. The learning problem is to determine parameterized function $\hat{F}(\mathbf{a})$, which approximately describes function F. Function $\hat{F}(\mathbf{a})$ characterizes the model of decision rule. Being under the "soft" classification, the randomized model occurs as a model of decision rule, i.e., it has randomized parameters $\mathbf{a}$. Its input are vectors $\{\mathbf{e}^{(1)}, \ldots, \mathbf{e}^{(h)}\}$, and the output $\hat{\mathbf{y}}(\mathbf{a})$ depends on randomized parameters of $\mathbf{a}$. As such, we choose a model of a single-layer neural net [21]:

$$\hat{y}^{(i)}(\mathbf{a}) = \text{sigm}\left(\langle \mathbf{e}^{(i)}, \mathbf{b} \rangle\right), \qquad i = \overline{1, n}, \tag{2}$$

where

$$\begin{aligned} \text{sigm}(x_i) &= \frac{1}{1 + \exp[-\alpha(x_i - \Delta)]}, \\ x_i &= \left(\langle \mathbf{e}^{(i)}, \mathbf{b} \rangle\right), \\ \mathbf{a} &= \{\mathbf{b}, \alpha, \Delta\}. \end{aligned} \tag{3}$$

Figure 1 shows a graph of the sigmoid function with parameters: "slope" α and "threshold" Δ. The function of $\mathrm{sigm}(x)$ in the interval $[1/2, 1]$ corresponds to the first class and the values in the interval $[0, 1/2)$-to the second class.

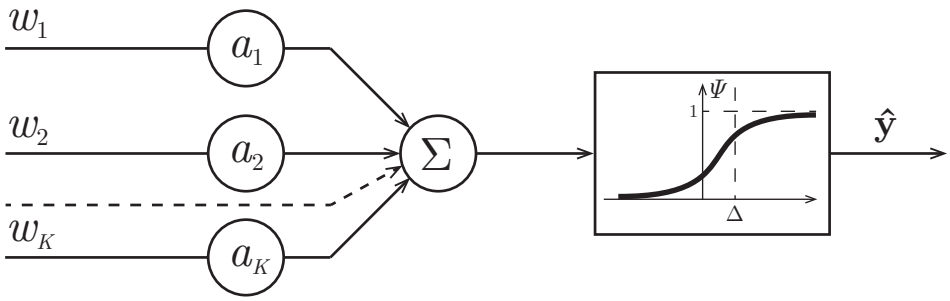

Figure 1. Structure of a single-layer neural net.

In the randomized Models (2) and (3) parameters $\mathbf{a} = \{a_1, \ldots, a_{(n+2)}\}$ are of interval type:

$$\begin{aligned} a_k &\in \mathcal{A}_k = [a_k^-, a_k^+], \qquad k = \overline{1, n+2}, \\ \mathcal{A} &= \bigotimes_{k=1}^{n+2} \mathcal{A}_k. \end{aligned} \tag{4}$$

Their probabilistic properties are characterized by *PDF* $P(\mathbf{a})$, which is defined over set $\mathcal{A}$.

Since parameters $\mathbf{a}$ are random, then for each object e_i, an ensemble $\hat{\mathcal{Y}}^{(i)}$ of random numbers $\hat{y}^{(i)}(\mathbf{a})$ occurs for the interval $[0, 1]$. We define it as an average:

$$\mathcal{M}\{\hat{y}^i(\mathbf{a})\} = \int_{\mathcal{A}} P(\mathbf{a}) \mathrm{sigm}\left(\langle \mathbf{e}^{(i)}, \mathbf{a} \rangle\right) d\mathbf{a}, \qquad i = \overline{1, h}. \tag{5}$$

Therefore, according to *RML* [16] general procedure, the problem of "2"-soft classification is represented as follows:

$$\mathcal{H}[P(\mathbf{a})] = -\int_{\mathcal{A}} P(\mathbf{a}) \ln P(\mathbf{a}) d\mathbf{a} \Rightarrow \max, \tag{6}$$

when

$$\int_{\mathcal{A}} P(\mathbf{a}) d\mathbf{a} = 1, \tag{7}$$

$$\int_{\mathcal{A}} P(\mathbf{a}) \mathrm{sigm}\left(\langle \mathbf{e}^{(i)}, \mathbf{a} \rangle\right) d\mathbf{a} = y_i, \qquad i = \overline{1, h}, \tag{8}$$

where $P(\mathbf{a})$ belongs to class $\mathbb{C}^1$ of continuously differentiable functions. This is a problem of functional entropy-linear programming, which has an analytical solution, such as entropy optimal *PDF* $P^*(\mathbf{a} \mid \theta)$, parameterized by Lagrange multipliers θ:

$$P^*(\mathbf{a} \mid \theta) = \frac{\exp\left[-\sum_{i=1}^h \theta_i \hat{y}^{(i)}(\mathbf{a})\right]}{\mathcal{P}(\theta)}, \tag{9}$$

where $\hat{y}^{(i)}(\mathbf{a})$ is defined by the Equality (2), and

$$\mathcal{P}(\theta) = \int_{\mathcal{A}} \exp\left[-\sum_{i=1}^h \theta_i \hat{y}^{(i)}(\mathbf{a})\right]. \tag{10}$$

Lagrange multipliers are determined from the Equation (8).

2.2. Testing

At this point, a collection of objects $\mathcal{T}$ is used, which are characterized by the vector of responses $\mathbf{z} = \{z_1, \ldots, z_r\}$ with known objects belonging to grade 1 or 2. Vector $\mathbf{z}$ will be used to evaluate the quality of testing. In the testing procedure itself, only $\mathcal{T}$ objects collection will be used.

The subject of the testing is randomized decision Rules (2) and (3) with entropy optimal *PDF* function of parameters. At the same time, a trial sequence of Monte Carlo is implemented with volume N, every one of which is generated by a random vector $\mathbf{a}$ with appropriate entropy optimal *PDF* function $P^*(\mathbf{a})$ (6)–(8). Assume that, as a result of these tests, it was found that the first object from the testing collection was assigned to the first class N_1 times and $N - N_1$ times to the second class; .... k-th object was assigned to the first class N_k times and $(N - N_k)$–to the second class, etc. For a sufficiently large number of tests, the empirical probability can be determined

$$\begin{array}{rcl} p_1^{(1)} & = & \frac{N_1}{N}, \ldots, p_1^{(k)} = \frac{N_k}{N}, \ldots, \ldots \\ p_2^{(1)} & = & \frac{N-N_1}{N}, \ldots, p_2^{(k)} = \frac{N-N_k}{N}, \ldots, \ldots. \end{array} \tag{11}$$

Therefore, the testing algorithm can be represented as follows (where i is the number of the object):

Step 1-i. In accordance with the optimal *PDF* function, a set of output values is generated with entropy optimal Models (2) and (3), comprising N random numbers from an interval $[0, 1]$.

Step 2-i. If the random number from this set is larger than $1/2$, then the object $t^{(i)}$ belongs to the class 1. If it is less than $1/2$, then it belongs to class 2.

Step 3-i Empirical probabilities are determined (11).

As a result of the functioning of this procedure, any object can be defined in one of two classes with a certain probability, which reflects an uncertainty within data and models of decision rule.

The transition to a hard classification can be accomplished by fixing the threshold of probabilities, an object which above it belongs to the relevant class. The number of objects that can be "hard"-classified depends on the threshold value. It is not difficult to find that when thresholds are greater than 0.5, not all objects are classified, but with more than 0.5 probability. However, at thresholds less than 0.5, all are classified, but with less than 0.5 probability.

3. Model Examples of "2"-Soft Classification

In this section, we present the model experiments conducted in accordance with the proposed learning algorithm. It should be noted that all data sets are synthetic and generated manually with a standard random number generator. The first series of experiments aims to introduce the proposed computational procedure and should be considered as illustrative examples. On the other hand, the last example in the next section is more important and demonstrates the advantages of the probabilistic approach for classification in the presence of data errors.

3.1. Soft "2"-Classification of Four-Dimensional Objects

Consider that the objects characterized by four features are coordinates of vectors $\mathbf{e}$ and $\mathbf{t}$.

3.1.1. Learning

The learning collection consists of three objects, every one of which is described by four attributes, and the values of which are shown in Table 1.

Table 1. Learning data example.

i	$e_1^{(i)}$	$e_2^{(i)}$	$e_3^{(i)}$	$e_4^{(i)}$
1	0.11	0.75	0.08	0.21
2	0.91	0.65	0.11	0.81
3	0.57	0.17	0.31	0.91

Randomized model of decision Rules (2) and (3) has parameters: $\alpha = 1.0$ and $\Delta = 0$. Learning vector ("teacher's" answers) $\mathbf{y} = \{0.18; 0.81; 0.43\}$ ($y_i < 0.5$ corresponds. to class 2; $y_i \geq 0.5$ corresponds to class 1). Lagrange multipliers for the entropy-optimal *PDF* (9) have the following values: $\bar{\theta}^* = \{0.2524; 1.7678; 1.6563\}$. Parameters $a_i \in [-10, 10]$, $i = \overline{1,4}$. Entropy-optimal *PDF* function for this learning collection is as follows:

$$P^*(\mathbf{a}, \bar{\theta}) = \frac{\exp\left(-\sum_{i=1}^{3} \theta_i y_i(\mathbf{a})\right)}{\mathcal{P}(\bar{\theta})}, \tag{12}$$

$$y_i(\mathbf{a}) = \left(1 + \exp(-\sum_{k=1}^{4} e_k^{(i)}, a_k)\right)^{(-1)}.$$

Figure 2 shows a two-dimensional section *PDF* $P^*(\mathbf{a}, \bar{\theta}^*)$.

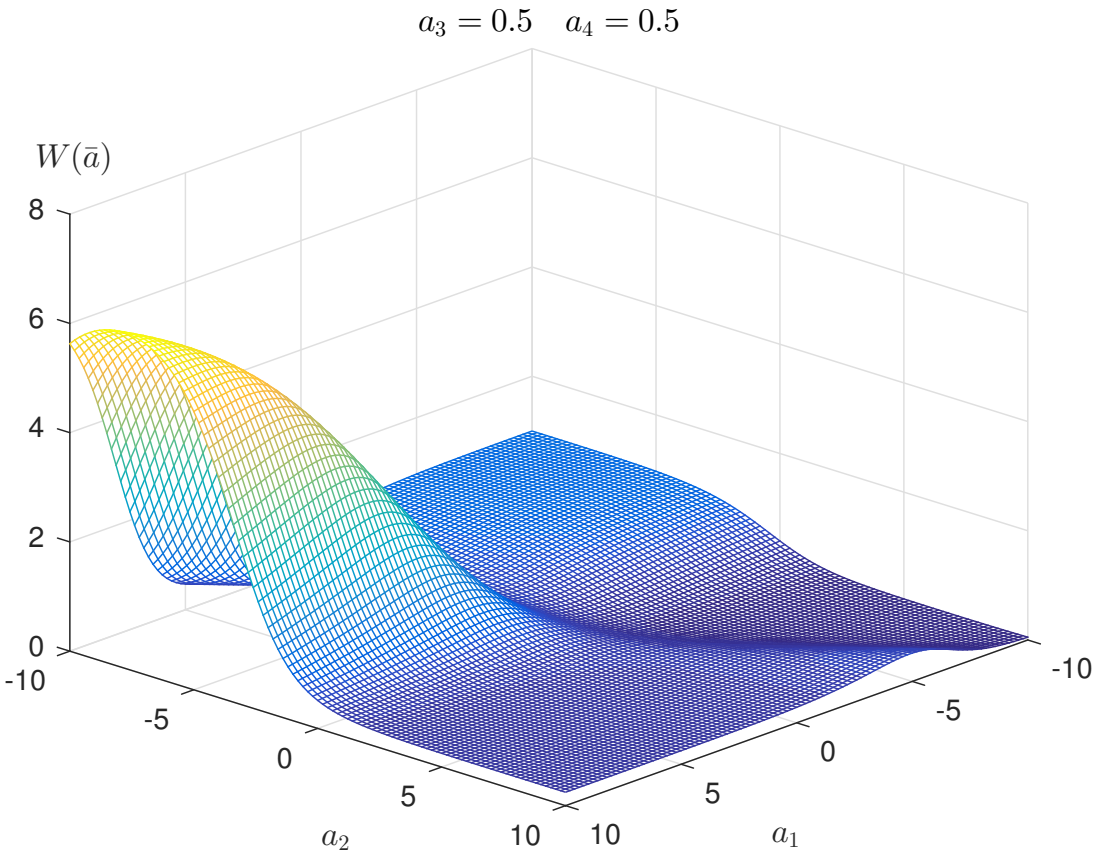

Figure 2. Two-dimensional section of probability density function (*PDF*) $P^*(\mathbf{a}, \bar{\theta}^*)$.

3.1.2. Testing

At this stage, a set of objects $\mathbb{T} = \{t_1, \ldots, t_r\}$ is used, where each element of the set is characterized by vector $\mathbf{t}^{(j)} \in R^{(4)}$. The generated array of (500×4) four-dimensional random vectors $\mathbf{t}^{(i)}$, $i = \overline{1,500}$, with independent components evenly distributed in intervals $[0, 1]$. Then, the algorithm of "2"-soft classification is applicable. Figure 3 shows the empirical probabilities $p_1^{(i)}, p_2^{(i)}$ of belonging of t_i-object to Classes 1 and 2.

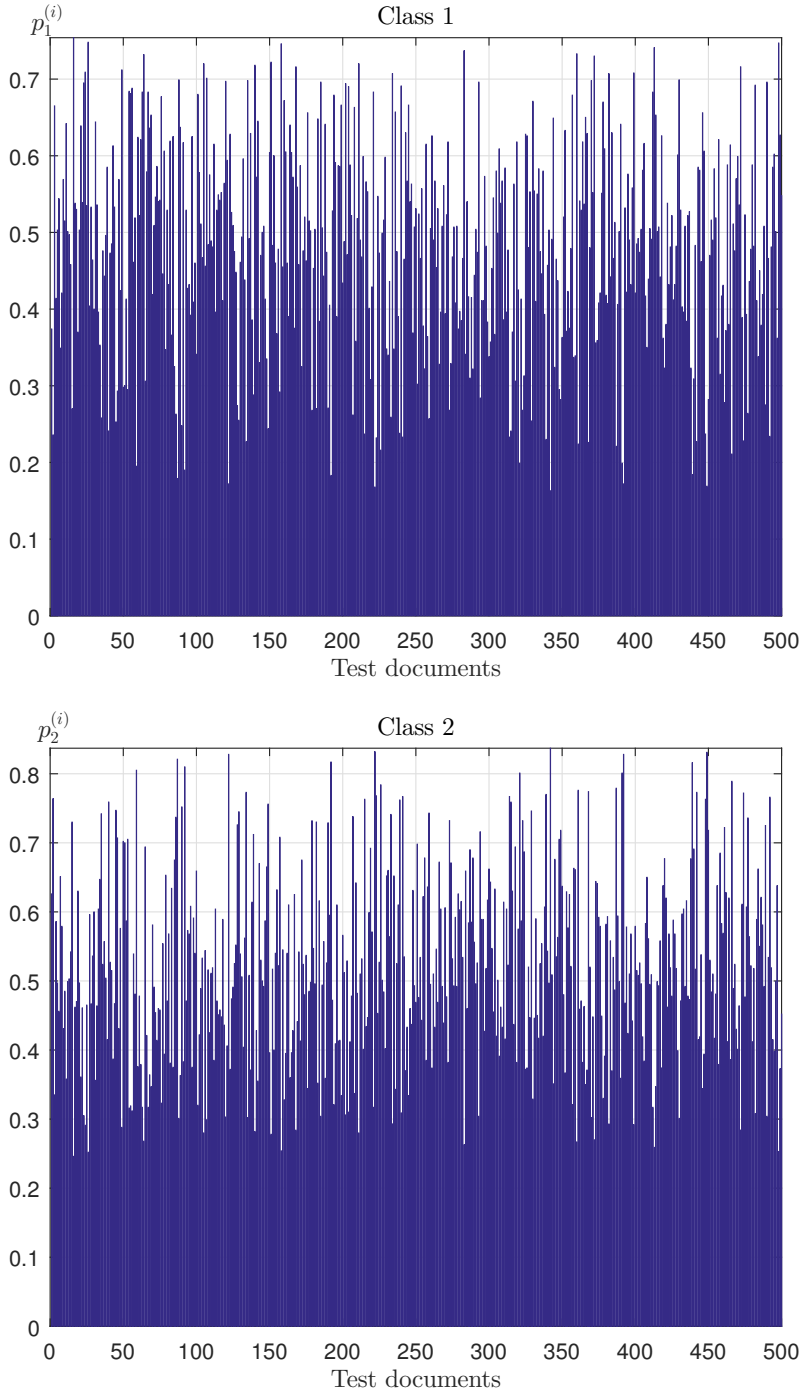

Figure 3. The empirical probabilities for example 1.

3.2. Two-Dimensional Objects "2"-Soft Classification

Consider the objects characterized by two features that are coordinates of the vectors **e** and **t**.

3.2.1. Learning

The learning collection consists of three objects, every one of which is described by two attributes, and the values of which are shown in Table 1. The values of parameters α, Δ and intervals for random parameters **a** correspond to Example 1. Lagrange multipliers for the entropy-optimal *PDF* (9) have the

following values: $\bar{\theta}^* = \{9.6316; -18.5996; 16.7502\}$. Entropy-optimal *PDF* function $P^*(\mathbf{a} \mid \bar{\theta})$ for this learning collection is as follows:

$$\begin{aligned} P^*(\mathbf{a}) &= \frac{\exp\left(-\sum_{i=1}^{3} \theta_i y_i(\mathbf{a})\right)}{\mathcal{P}(\bar{\theta})}, \\ y_i(\mathbf{a}) &= \left(1 + \exp(-\sum_{k=1}^{2} e(i)_k, a_k)\right)^{(-1)}. \end{aligned} \tag{13}$$

Figure 4 shows function $P^*(\mathbf{a}, \bar{\theta}^*)$.

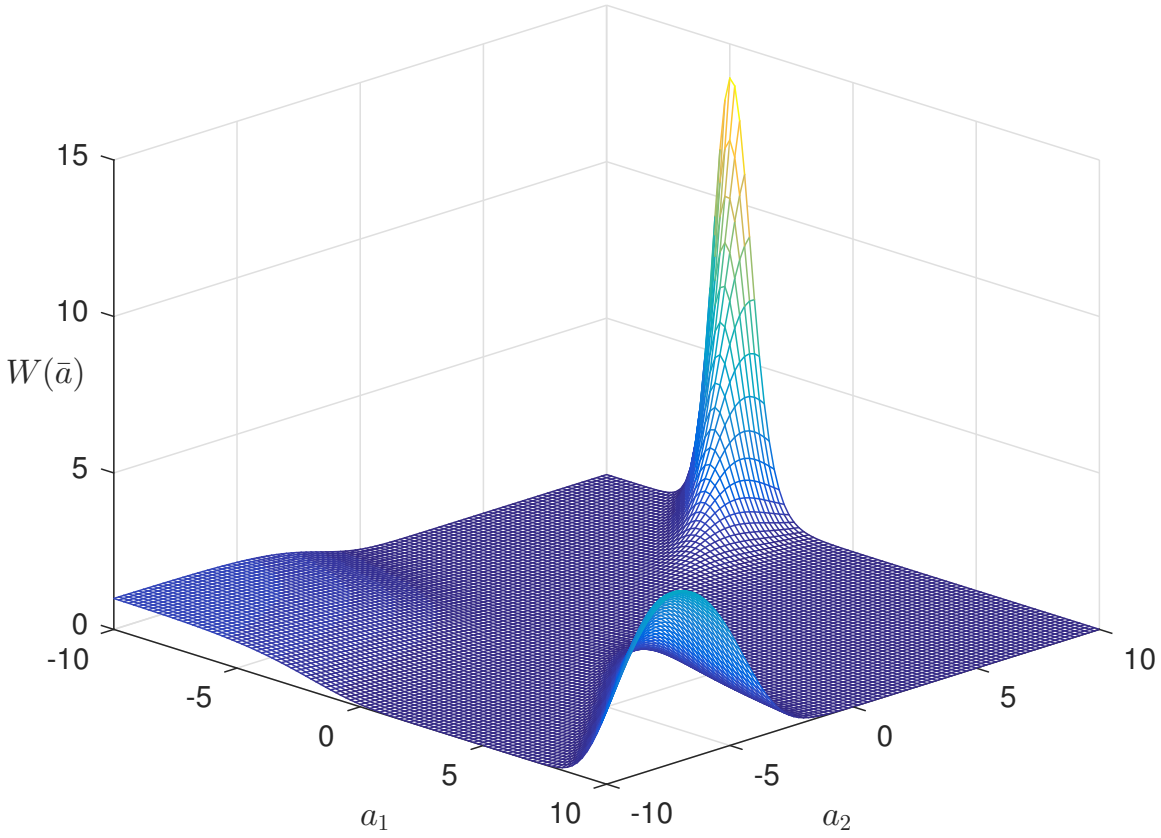

Figure 4. Entropy-optimal *PDF* function $P^*(\mathbf{a}, \bar{\theta}^*)$.

3.2.2. Testing

All parameters of this example correspond to Example 2. Figure 5 shows empirical probabilities $p_1^{(i)}, p_2^{(i)}$ of belonging of the t_i-object to Classes 1 and 2 ($i = \overline{1, 500}$).

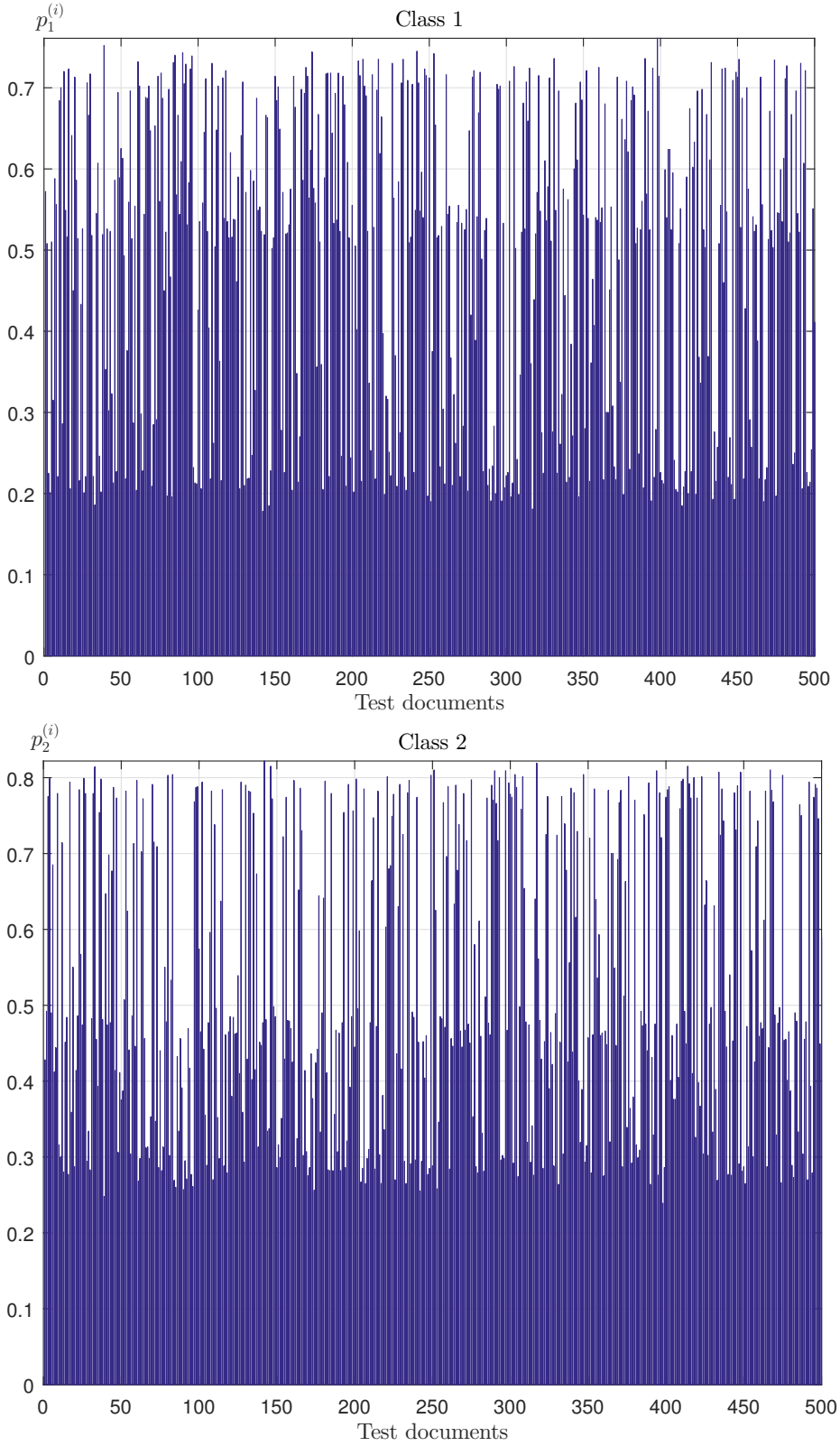

Figure 5. The empirical probabilities for example 2.

4. Experimental Studies of "2"-Hard/Soft Classifications in Presence of Data Errors

"2"-soft classification, based on entropy randomization decision rules, creates a family of "2"-hard classifications, parameterized by thresholds of belonging probabilities. This family complements "2"-hard classifications based on machine learning methods, particularly using the method of least squares [9].

4.1. Data

The experimental study of the soft and hard classifications was performed on simulated data with model errors. All the data for the objects were labeled in order of belonging to one of two classes. This information was used for learning the model of decision rule, and during the method testing to evaluate the classification accuracy.

The objects of classification are characterized by four-dimensional vectors:

$$\mathbf{u}^{(i)} = \{u_1^{(i)}, \dots, u_4^{(i)}\}, \qquad i = \overline{1,N}, \quad N = 510. \tag{14}$$

Vectors of both collections were chosen in a random order, and they were evenly distributed on a four-dimensional unit cube. Data errors are modeled by random and evenly distributed four-dimensional vectors $\bar{\xi} \in [-\bar{\xi}^{-}, \bar{\xi}^{+}]$, where $\bar{\xi}^{\pm} = \pm 0.1$.

To mark the belonging of vectors $\bar{u}$ to one of the two classes, the number generation procedure from interval $[0,1]$ is applied as follows:

$$y^{(i)} = sigm\left(\sum_{s=1}^{4} e_s^{(i)} b_s^0\right) + \xi^{(i)}, \qquad i = \overline{1,510}, \tag{15}$$

where

$$sigm(x) = \frac{1}{1+\exp(-\alpha(x-\Delta))}, \ \mathbf{b}^0 = \{0.1, 0.2, 0.3, 0.4\}, \ \alpha = 3, \ \Delta = 0.5. \tag{16}$$

Let us recall that the values of the sigmoid function from interval $[0.5, 1]$ correspond to the first class, and from interval $[0, 0.5)$ to the second class, i.e.,

$$c_i = \begin{cases} 1, \text{for } 0.5 \le y^{(i)} \le 1, \\ 2, \text{for } 0.0 \le y^{(i)} < 0.5. \end{cases} \tag{17}$$

Thus, numerical characteristics of objects components of u_s^i, $s = \overline{1,4}$; $i = \overline{1,510}$ vectors serve as their features, and c_i values refer them to a certain class.

Two collections were formed from these objects: the learning one $\mathcal{E} = \{\mathbf{e}^{(1)}, \dots, \mathbf{e}^{(10)}\}$ and the testing one $\mathcal{T} = \{t^{(1)}, \dots, t^{(500)}\}$.

4.2. Randomized Model (Decision Rule)

In the experimental study, we used a model that coincides by its structure with (15) and (16), but has randomized parameters and noises:

$$\hat{y}^{(i)}(\mathbf{a}) = sigm\left(\sum_{s=1}^{4} e_s^{(i)} a_s\right) + \zeta^{(i)}, \qquad i = \overline{1,10}, \tag{18}$$

where $\mathbf{a} = \{a_1, \dots, a_4\}$ are randomized parameters of the interval type. For example, $\mathbf{a} \in \mathcal{A} = [-1, 1]$; $\zeta^{(i)}$ is a random interval type noise, i.e., $\zeta^{(i)} \in \mathcal{K}_i = [-0.1, 0.1]$, $i = \overline{1,10}$. Noise probabilities $\zeta^{(i)}$ are characterized by *PDF* $q_i(\zeta^{(i)})$, which exists as a continuously differentiable function.

The optimization problem of entropy-robust estimation is formulated as follows [16]:

$$H[P(\mathbf{a}), Q(\xi)] = -\int_{\mathcal{A}} P(\mathbf{a}) \ln P(\mathbf{a}) d\mathbf{a} - \sum_{i=1}^{10} \int_{\mathcal{K}_i} q_i(\zeta^{(i)}) \ln q_i(\zeta^{(i)}) d\zeta^{(i)} \Longrightarrow \max_{P,Q} \tag{19}$$

under constraints:

$$\int P(\mathbf{a}) d\mathbf{a} = 1, \qquad \int_{\mathcal{K}_i} q_i(\zeta^{(i)}) d\zeta^{(i)} = 1, \quad i = \overline{1,10}; \tag{20}$$

and

$$\mathcal{M}[\hat{y}^{(i)}(\mathbf{a})] = \int_{\mathcal{A}} P(\mathbf{a}) \hat{y}^{(i)}(\mathbf{a}) d\mathbf{a} + \int_{\mathcal{K}_i} q_i(\zeta^{(i)}) d\zeta^{(i)} = y^{(i)}, \quad i = \overline{1,10}. \tag{21}$$

The problem (19)–(21) has an analytical solution in the form of

$$P^*(\mathbf{a}, \bar{\theta}) = G^{-1}(\theta)\, W(\mathbf{a}, \bar{\theta}), \tag{22}$$

$$W(\mathbf{a}, \bar{\theta}) = \exp\left(\sum_{i=1}^{10} \theta_i \hat{y}^{(i)}(\mathbf{a})\right), \qquad G(\theta) = \int_A \exp\left(\sum_{i=1}^{10} \theta_i \hat{y}^{(i)}(\mathbf{a})\right) d\mathbf{a}, \tag{23}$$

and

$$q_i^*(\zeta^{(i)}, \theta_i) = F_i^{-1}(\theta_i)\, U_i(\zeta^{(i)}, \theta_i), \tag{24}$$

$$U_i(\zeta^{(i)}, \theta_i) = \exp\left(\theta_i \zeta^{(i)}\right), \qquad F_i(\theta_i) = \int_{\mathcal{K}_i} \exp\left(\theta_i \zeta^{(i)}\right) d\zeta^{(i)}, \tag{25}$$

where $\theta = \{\theta_1, \ldots, \theta_{10}\}$ are Lagrange multipliers.

In this example, the optimization was performed by *MATLAB* 2015b software (Version 8.6, Build 267246), using the optimization tools (optimization toolbox) and statistical learning (statistic and machine learning toolbox). The values obtained for Lagrange multipliers are: $\bar{\theta}^* = 10^{-3} \times \{0.1684, 0.0275, 0.0288, 0.0110, 0.0326, 0.0429, -0.0469, 0.0073, -0.0012, -0.0354\}$.

Figure 6 shows a two-dimensional section of entropy-optimal *PDF* function $P^*(\mathbf{a}, \bar{\theta}^*)$.

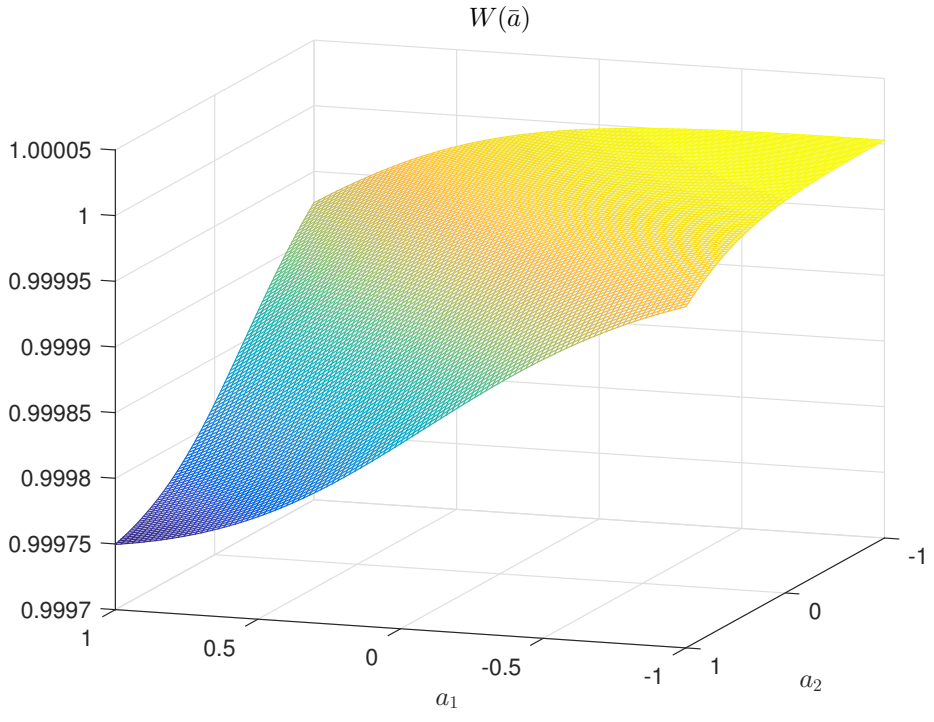

Figure 6. Two-dimensional section of function $P^*(\mathbf{a}, \bar{\theta}^*)$.

4.3. Testing of Learning Model: Implementation of "2"-Soft Classification

The process of randomized classified algorithm testing involves a generation of a random set of model parameters according to received entropy-optimal density functions. To generate an ensemble of random values, an algorithm of Metropolis–Hastings is used [22].

In this example, the vectors set was generated $[\mathbf{a}^{(k)}]^* = \{[a_1^{(k)}]^*, \ldots, [a_4^{(k)}]^*\}, k = \overline{1,100}$, with entropy-optimal *PDF* $P^*(\mathbf{a}, \bar{\theta}^*)$ (22), defining 100 implementations of decision rule classification models. For each object (i) from the testing collection, we obtain an ensemble of numbers from the interval $[0,1]$

$$\hat{y}_k^{(i)}([\mathbf{a}^{(k)}]^*) = sigm\left(\sum_{s=1}^{4} e_s^{(i)}[a_s^{(k)}]^*\right), \qquad k = \overline{1,100}. \tag{26}$$

Along this ensemble, the objects are distributed into classes in accordance with rule

$$c^{(i)} = \begin{cases} 1, \text{for } 0.5 \le \hat{y}^{(i)} \le 1, \\ 2, \text{for } 0.0 \le \hat{y}^{(i)} < 0.5. \end{cases} \tag{27}$$

The probabilities of belonging to Classes 1 and 2 are calculated as follows:

$$p_1^{(i)} = \frac{N_1[c_k^{(i)} == 1]}{K}, \quad p_2^{(i)} = \frac{N_2[c_k^{(i)} == 2]}{K}, \qquad K = 100, \tag{28}$$

Here, N_1 and N_2 are the numbers of tests in which the objects were classified as Classes 1 and 2, respectively. Figure 7 shows the value distribution of empirical probability of belonging to Classes 1 and 2 for testing sample objects, which characterize "2"-soft classification with entropy optimal linear decision rule.

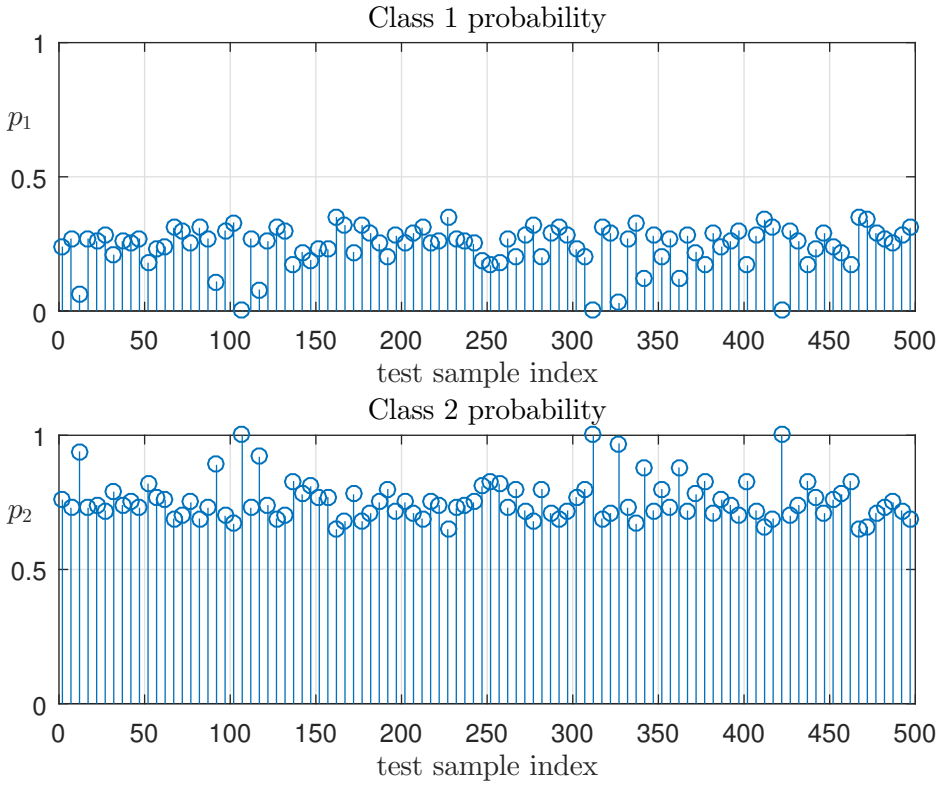

Figure 7. The empirical probabilities of belonging to classes 1 and 2.

It was noted above that with "2"-soft classification, you can generate families of "2"-hard classification, assigning various thresholds η of empirical probability. Belonging to a class is defined by the following condition:

$$\text{Object } (i) \in \text{class 1, if } p_1^i \geq \delta. \tag{29}$$

Let us determine the accuracy of classification as:

$$\eta = \frac{L_{tr}}{L} 100\%, \tag{30}$$

where L is the length of the test collection, and L_{tr} is the number of correct classifications.

The research of the above example shows the existence of dependence between the accuracy η and the threshold δ. Figure 8 shows this dependence, where we can see that the quantity η attains a maximum value equal to 78.5% at $\delta = 0.19$. However, "2" hard classification, which uses the decision rule (26) with non-randomized parameters and the method of least squares to determine their values, yields a 66% accuracy (regardless of the threshold).

This result means that a traditional exponential model widely used for classification problems [6,9] could be improved by statistical interpretation of its output. In this case, the proposed randomized machine learning technique that evolves the generation of entropy-optimal probability functions, and variation of soft decision threshold boosts the accuracy of classification to nearly 80 percent.

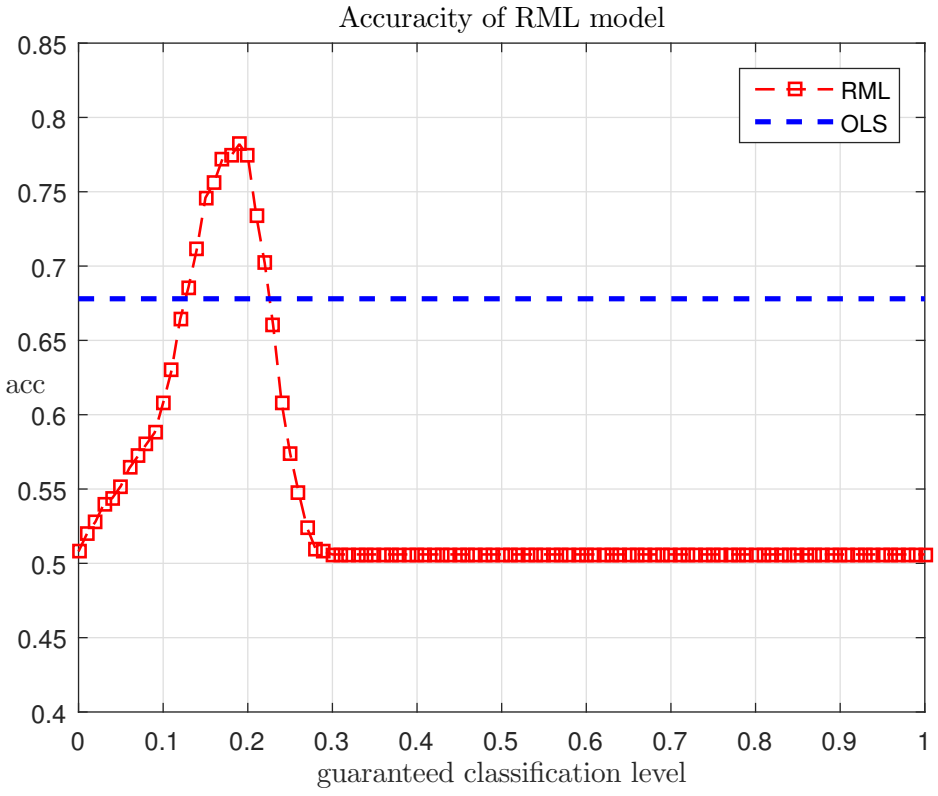

Figure 8. Dependence of *RML* model accuracy on the threshold δ.

5. Conclusions

A method for "2"-soft classification was proposed, which allows for referring objects with calculated empirical probability to one of two classes. The latter is determined by the Monte Carlo method with the use of the entropy-optimal randomized model of the decision rule. A corresponding

problem is formulated for the maximization of entropy functional on the set, a configuration of which is determined by balances between the real input data and average output of the randomized model of the decision rule.

The problem of "2"-soft classification for the case of existence of data errors simulated by the additive noise evenly distributed in the parallelogram. The entropy-optimal estimations of probability distribution density for model parameters and for noises, which are the best at the maximum indeterminateness in terms of entropy, were obtained. We performed an experimental comparison of "2"-hard and "2"-soft classifications. An existence of the classification threshold interval was revealed, whereby a precision of "2"-soft classification (the number of correct answers) was increased by 20. Examples illustrating the proposed method are provided.

Acknowledgments: This work was supported by the Russian Foundation for Basic Research (Project No. 16-07-00743). We also thank the comments from the two anonymous reviewers, which improved the quality of the paper.

Author Contributions: Yuri S. Popkov and Zeev Volkovich introduced the ideas of this research, formulated the main propositions and wrote the paper; Yuri A. Dubnov made the experiments; and Renata Avros and Elena Ravve reviewed the paper and provided useful comments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Rosenblatt, M. *The Perceptron—Perceiving and Recognizing Automaton*; Report 85-460-1, 1957. Available online: http://blogs.umass.edu/brain-wars/files/2016/03/rosenblatt-1957.pdf (accessed on 19 April 2017)
2. Tsipkin, Y.Z. *Basic Theory of Learning Systems*; Nauka (Science): Moscow, Russia, 1970.
3. Ayzerman, M.A.; Braverman, E.M.; Rozonoer, L.I. *A Potential Method of Machine Functions in Learning Theory*; Nauka (Science): Moscow, Russia, 1970.
4. Vapnik, V.N.; Chervonenkis, A.Y. *A Theory of Pattern Recognition*; Nauka (Science): Moscow, Russia, 1974.
5. Vapnik, V.N.; Chervonenkis, A.Y. *A Recovery of Dependencies for Empirical Data*; Nauka (Science): Moscow, Russia, 1979.
6. Bishop, C.M. *Pattern Recognition and Machine Learning*; Series: Information Theory and Statistics; Springer: New York, NY, USA, 2006.
7. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer, 2009. Available online: https://statweb.stanford.edu/~tibs/ElemStatLearn/ (accessed on 19 April 2017).
8. Merkov, A.B. *Pattern Recognition. Building and Learning Probabilistic Models*; M. LENAND: Moscow, Russia, 2014.
9. Vorontsov, K.V. *Mathematical Methods of Learning by Precedents*; The Course of Lectures at MIPT; Moscow, Russia, 2006. Available online: http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf (accessed on 19 April 2017)
10. Zolotykh, N.Y. Machine Learning and Data Analysis. 2013. Available online: http://www.uic.unn.ru/~zny/ml/ (accessed on 19 April 2017).
11. Boucheron, S.; Bousquet, O.; Lugosi, G. Theory of Classification: A Survey of Some Recent Advances. *ESAIM Probab. Stat.* **2005**, *9*, 323–375. Available online: http://www.esaim-ps.org/articles/ps/pdf/2005/01/ps0420.pdf (accessed on 19 April 2017).
12. Smola, A.; Bartlett, P.; Scholkopf, B.; Schuurmans, D. *Advances in Large Margin Classifiers*; MIT Press: Cambridge, MA, USA, 2000.
13. Jain, A.; Murty, M.; Flunn, P. Data Clustering: A Review. *ASM Comput. Surv.* **1999**, *31*, 264–323.
14. Brown, G. Ensemble learning. In *Encyclopedia of Machine Learning*; Sammut, C., Webb, G.I., Eds.; Springer: New York, NY, USA, 2010; pp. 312–320.
15. Furnkranz, J.; Gamberger, D.; Lavrac, N. *Foundations of Rule Learning*; Springer: Berlin/Heidelberg, Germany, 2012.
16. Popkov, Y.S.; Dubnov, Y.A.; Popkov, A.Y. Randomized Machini Learning: Statement, Solution, Applications. In Proceedings of the IEEE International Conference on Intelligent Systems, Sofia, Bulgaria, 4–6 September 2016.

17. Kamal, N.; John, L.; Andrew, M. Using Maximum Entropy for Text Classification. In *IJCAI-99 Workshop on Machine Learning for Information Filtering*. Available online: http://www.cc.gatech.edu/~isbell/reading/papers/maxenttext.pdf (accessed on 19 April 2017).
18. Payton, L.; Fu, S.-W.; Wang, S.-S.; Lai, Y.-H.; Tsao, Y. Maximum Entropy Learning with Deep Belief Networks. *Entropy* **2016**, *18*, 251.
19. Amos, G.; George, G.; Judge, D.M. *Maximum Entropy Econometrics: Robust Estimation with Limited Data*; John Wiley and Sons Ltd.: Chichester, UK, 1996; p. 324.
20. Japkowicz, N.; Shah, M. *Evaluating Learning Algorithms: A Classification Perspective*; Cambridge University Press: Cambridge, UK, 2011.
21. Gerstner, W.; Kishler, W.M. *Spiking Neuron Models: Single Neurons, Population, Plasticity*; Cambridge University Press: Cambridge, UK, 2002.
22. Rubinstein, R.Y.; Kroese, D.P. *Simulation and Monte Carlo Method*; John Willey and Sons: Hoboken, NJ, USA, 2008.