

COMPARATIVE ANALYSIS OF CLASSIFIER FUSERS

Marcin Zmyslony¹, Michal Wozniak¹, and Konrad Jackowski

¹ Department of Systems and Computer Networks, Wroclaw University of Technology

Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, Poland

[marcin.zmyslony, michal.wozniak, konrad.jackowski]@pwr.wroc.pl

ABSTRACT

There are many methods of decision making by an ensemble of classifiers. The most popular are methods that have their origin in voting method, where the decision of the common classifier is a combination of individual classifiers' outputs. This work presents comparative analysis of some classifier fusion methods based on weighted voting of classifiers' responses and combination of classifiers' discriminant functions. We discuss different methods of producing combined classifiers based on weights. We show that it is not possible to obtain classifier better than an abstract model of committee known as an Oracle if it is based only on weighted voting but models based on discriminant function or classifier using feature values and class numbers could outperform the Oracle as well. Delivered conclusions are confirmed by the results of computer experiments carried out on benchmark and computer generated data.

KEYWORDS

Pattern recognition, combined classifier, classifier ensemble, classifier fusion, multiple classifier systems

1. INTRODUCTION

Problem of pattern recognition is accompanying our whole life. People cannot even imagine how many classification problems are being solved in each second of their regular day. The trivets example could be morning breakfast where human's brain does classification for: temperature of coffee or tea, food to eat, TV station to watch and so many others factors which could have influence on our life. The main problem of the computers and software dedicated for them, is that they do not have intelligence and for each classification task they have to use algorithms (methods) which give very close results to that, which are generated by human's brain. Therefore methods of automatic pattern recognition are the main trends in Artificial Intelligence. The aim of such task is to classify the object to one of predefined categories, on the basis of observation of its features [5]. Such methods are applied to many practical areas like prediction of customer behaviour [2, 4], fraud detection [22], medical diagnosis [23] etc. Numerous approaches have been proposed to construct efficient, high quality classifiers like neural networks, statistical and symbolic learning [24]. There are much current researches into developing even more efficient and accurate recognition algorithms. Among them the multiple classifier systems are currently the focus of intense research. The subject matter has been known for over 15 years [28]. Some works in this field were published as early as the '60 of the XX century [3], when it was shown that the common decision of independent classifiers is optimal, when chosen weights are inversely proportional to errors made by the classifiers. In many review articles this trend has

been mentioned as one of the most promising in the field of the pattern recognition [14]. In the beginning in literature one could find only majority vote, but in later works more advanced methods of finding a common solution to the classifier group problem were proposed. Estimation accuracy of the classifier committee is one of fundamental

importance. Known conclusions, derived on analytic way, concern particular case of the majority vote [10] when classifier committee is formed on the basis of independent classifiers. Unfortunately this case has only theoretical character and is not useful in practice. The weighted voting is taken into consideration [9, 11, 19, 24], but a problem of establishing weights for mentioned voting procedure is not simple. Many of authors have proposed treating the voting block as a kind of classifier [12] but the general question is “does the fuser need to be trained?” [7]. An alternative way of common classifier construction is combination of discriminant functions of available classifiers in order to obtain set of common discriminant functions, e.g. via linear combination.

Paper presents comparative analysis of some methods of classifier fusion based on weighted voting of classifiers’ responses and combination of classifiers’ discriminant functions. It is also discuss which of presented methods could produce classifier better than the abstract model of classifier committee known as Oracle.

2. MODELS OF WEIGHTED VOTING

The *Oracle* is an abstract fusion model, which recognizes object correctly, if at last one of the classifiers in committee points at correct class too. It is usually used in comparative experiments to show limits of classifier committee quality [24]. But the key question is if the *Oracle* is really such a limit. In this work we will consider the question of whether it is possible to obtain a compound classifier capable of achieving higher accuracy than the *Oracle* one. Let us consider some methods of classifier fusion on the basis of classifiers’ response and discriminant function they used then we will consider which of presented fusion methods could produce better classifier than *Oracle*.

2.1. Classifier fusion based on classifier response

Let us assume that we have n classifiers $\Psi^{(1)}, \Psi^{(2)}, \dots, \Psi^{(n)}$, which decides if object belongs to class $i \in M = \{1, \dots, M\}$. For making common decision by the group of classifiers we use following common classifier $\bar{\Psi}$:

$$\bar{\Psi}(x) = \arg \max_{j \in M} \sum_{l=1}^n \delta(j, \Psi^{(l)}(x)) w^{(l)}, \quad (1)$$

where $w^{(l)}$ is the weight of the l -th classifier and

$$\delta(j, i) = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases}. \quad (2)$$

Let us note that $w^{(l)}$ plays key-role of the quality of classifier $\Psi^{(l)}$. There are many researches on how to set the weights. The examples are [10, 19] where authors proposed to learn the fuser.

Let us consider 3 possibilities of weight values set up:

- $w^{(l)}$ - weights are dependent on the classifier,
- $w^{(l)}(i)$ - weights are dependent on the classifier and the class number,
- $w^{(l)}(i, x)$ - weights are dependent on features values and assigned to the classifier and the class number.
- Let us note that for aforementioned models it is not possible to obtain classifier which is better than *Oracle* classifier because each decision rule making decision according to (1) could point to correct class if one classifier produces correct class label at least. Of course the best proposition is mentioned above *Oracle* classifier.
- The only model based (partial) on class label which could achieve better results than *Oracle* is classifier which produces decision on the basis of class labels given by $\Psi^{(1)}, \Psi^{(2)}, \dots, \Psi^{(n)}$ and feature vector values. That model was considered for example in [13, 20, 21, 26].
- **ILLUSTRATIVE EXAMPLE**
- Let us consider an exemplary pattern recognition problem with two classes and three elementary classifiers. Let us assume that the problem is discreet and entire feature space consists of six objects. The table 1. presents the results of recognition of all elementary classifiers we have got in hand (described as classifier 1, classifier 2, and classifier 3 respectively) along with their discrimination function (denoted as class 1 and class 2 support respectively) values calculated for six objects. Fails indicates if given decision is correct or not. For comparison purposes result obtained by *Oracle* is also presented in last column of the table.
- Table 1. Result of classification of elementary classifiers and Oracle classifier in given pattern recognition problem.

Object	correct class	Classifier 1				Classifier 2				Classifier 3				Oracle
		class 1 support	class 2 support	Decision	Fails	class 1 support	class 2 support	decision	fails	class 1 support	class 2 support	decision	fails	fails
1	1	0,68	0,32	1	0	0,51	0,49	1	0	0,98	0,02	1	0	0
2	1	0,45	0,55	2	1	0,12	0,88	2	1	0,96	0,04	1	0	0
3	2	0,48	0,52	2	0	0,99	0,01	1	1	0,47	0,53	2	0	0
4	1	0,49	0,51	2	1	0,87	0,13	2	1	0,48	0,52	2	1	1
5	1	0,47	0,53	2	1	0,99	0,01	2	1	0,47	0,53	2	1	1
6	2	0,05	0,95	2	0	0,58	0,42	2	0	0,15	0,85	2	0	0
Failures total					3				4				2	2

-

- The result of classifier measured with means of number of misclassifications (denoted as failures total) differs what shows some level of diversity among the elementary classifiers. This is positive effect as it can help improve result while fusing the classifier.
- Tables 2. Sets of common classifier's weights for three models of elementary classifiers' response fusion.

			Weights		
	object	Class	classifier 1	classifier 2	classifier 3
Response fusion model 1			0,1	0,8	0,1
Response fusion model 2		1	0,2	0,2	0,1
		2	0,3	0,8	0,1
Response fusion model 3	1	1	0,2	0,2	0,6
	1	2	0,2	0,8	0,6
	2	1	0,2	0,2	0,6
	2	2	0,2	0,2	0,6
	3	1	0,2	0,2	0,6
	3	2	0,2	0,8	0,6
	4	1	0,2	0,2	0,1
	4	2	0,2	0,8	0,1
	5	1	0,2	0,2	0,1
	5	2	0,2	0,8	0,1
6	1	0,2	0,2	0,1	
6	2	0,2	0,8	0,1	

In first fusion model the weights depend on classifiers only. In second one the class number affect the weights too while in the last model relation between the object and the weights has been established in addition.

The results of classification for three common classifier constructed on the basis of the elementary classifiers and the sets of weights depicted in Table 2. are presented in Table 3, Table 4, and Table 5. respectively.

Table 3. Results for classifier that fuses responses according to model 1.

		Elementary classifiers						Common classifier		Oracle
		classifier 1		classifier 2		classifier 3				
object	correct class	decision	weight	Decision	weight	decision	weight	decision	Fails	fails
1	1	1	0,1	1	0,8	1	0,1	1	0	0
2	1	2	0,1	2	0,8	1	0,1	2	1	0
3	2	2	0,1	1	0,8	2	0,1	1	1	0
4	1	2	0,1	2	0,8	2	0,1	2	1	1
5	1	2	0,1	2	0,8	2	0,1	2	1	1
6	2	2	0,1	2	0,8	2	0,1	2	0	0
failures total									4	2

Table 4. Results for classifier that fuses responses according to model 2.

		Elementary classifiers						Common classifier		Oracle
		classifier 1		classifier 2		classifier 3				
object	correct class	decision	weight	decision	weight	decision	weight	decision	fails	Fails
1	1	1	0,2	1	0,2	1	0,1	1	0	0
2	1	2	0,3	2	0,8	1	0,1	2	1	0
3	2	2	0,3	1	0,2	2	0,1	2	0	0
4	1	2	0,3	2	0,8	2	0,1	2	1	1
5	1	2	0,3	2	0,8	2	0,1	2	1	1
6	2	2	0,3	2	0,8	2	0,1	2	0	0
failures total									3	2

Table 5. Results for classifier that fuses responses according to model 3.

		Elementary classifiers						Common classifier		Oracle
		classifier 1		classifier 2		classifier 3				
object	correct class	decision	weight	decision	weight	decision	weight	decision	fails	Fails
1	1	1	0,2	1	0,2	1	0,6	1	0	0
2	1	2	0,2	2	0,2	1	0,6	1	0	0
3	2	2	0,2	1	0,2	2	0,6	2	0	0
4	1	2	0,2	2	0,8	2	0,1	2	1	1
5	1	2	0,2	2	0,8	2	0,1	2	1	1
6	2	2	0,2	2	0,8	2	0,1	2	0	0
failures total									2	2

The first model of response fusion is less flexible than the others and therefore finding appropriate weights for the classifiers is a relatively difficult task. It does not allow to fully exploit classifier local competences. The presented result is not impressive as 4 out of 6 objects are

misclassified. Better results were obtained for second model because impact of second classifier was reduced when it points at class 1. And finally the most flexible model that allows to evaluate classifier knowledge at different part of feature space gives the best results.

Naturally obtained results are only illustrative kind and we do not intend to claim that they prove general tendency.

2.2. Classifier fusion based on values of classifiers' discrimination function

This classification algorithm is formed by the procedures of classifier fusions on the basis of their discriminating function, the main form of which are *posterior* probability estimators, referring to the probabilistic model of a pattern recognition task [5]. The aggregating methods, which do not require learning, perform fusion with the help of simple operators such as maximum or average. However, they can be used in clearly defined conditions, as it has been presented in a research paper by Duin [7], which limits their practical applications. Weighting methods are an alternative and the selection of weights has a similar importance as it is in case of weighted majority voting. The advantages of this approach include an effective counteraction against the occurrence of elementary classifier overtraining.

Each classifier makes decision based on the value of discriminant function. Let $F^{(l)}(i, x)$ means such a function assigned to class i for given value of x , which is used by the l -th classifier $\Psi^{(l)}$. A common classifier $\hat{\Psi}(x)$ looks as follows

$$\hat{\Psi}(x) = i \quad \text{if} \quad \hat{F}(i, x) = \max_{k \in M} \hat{F}(k, x), \quad (3)$$

where

$$\hat{F}(i, x) = \sum_{l=1}^n w^{(l)} F^{(l)}(i, x) \quad \text{and} \quad \sum_{i=1}^n w^{(l)} = 1. \quad (4)$$

Varied interpretation of the discriminant function could be given. It could be the *posterior* probability for the classifiers based on Bayes decision theory or outputs of neural network. In general the value of such function means support given for distinguished class. Let us consider 4 possibilities of weight values set up:

- $w^{(l)}$ - weights are dependent on classifier,
- $w^{(l)}(i)$ - weights are dependent on classifier and class number,
- $w^{(l)}(x)$ - weights are dependent on classifier and feature vector,
- $w^{(l)}(i, x)$ - weights are dependent on classifier, class number, and feature vector.

Now let us consider which (and why) one of mentioned-above method could produce classifier better than *Oracle*. Firstly, let us note that for methods presented in points 1 and 3 such classifier could not achieve better quality than Oracle because it would mean that there is such a combination which produces correct decision when each classifier produces wrong one. For

mentioned cases if classifiers point at the wrong decision that meant that value of their discriminant functions assigned to the correct class is lower than assigned to the wrong ones. Let us consider case 3 which is more general than case 1. For such cases where weights' values are independent of class number, fuser produces wrong decision if

$$\sum_{l=1}^n w^{(l)}(x)F^{(l)}(correct_class, x) < \sum_{l=1}^n w^{(l)}(x)F^{(l)}(wrong_class, x). \quad (5)$$

It is not possible to set such values of weights for which presented relation be the inverse one because it is not possible via linear combination, but it is possible in the cases where weights are dependent additionally on class number.

2.3 Example of classifier fusion based on weights dependent on classifier and class number

Let us consider the same pattern recognition task that was introduced in previous section. Once again we have a set of elementary classifier presented in Table 1. This time we should focus on their discriminative functions (class support) as they will be a base for classifier fusion.

Table 6. presents the matrix weights used for fusion. It should be noticed that this time four fusion models are compared and therefore the table consists of four sections.

Table 6. Sets of classifier's weights for four models of elementary classifiers' discriminate function fusion.

			Weights		
	object	class	classifier 1	classifier 2	classifier 3
Discriminate function fusion model 1			0,2	0,8	0,1
Discriminate function fusion model 2		1	0,2	0,2	0,1
		2	0,3	0,8	0,2
Discriminate function fusion model 3	1		0,2	0,8	0,6
	2		0,2	0,8	0,6
	3		0,2	0,8	0,6
	4		0,8	0,2	0,1
	5		0,8	0,2	0,1
	6		0,8	0,2	0,1
Discriminate function fusion model 4	1	1	0,2	0,2	0,6
	1	2	0,2	0,8	0,6
	2	1	0,2	0,2	0,6
	2	2	0,2	0,2	0,6
	3	1	0,2	0,1	0,6
	3	2	0,2	0,8	0,6
	4	1	0,2	0,2	0,1
	4	2	0,2	0,8	0,1
	5	1	0,2	0,2	0,1
	5	2	0,2	0,8	0,1
6	1	0,2	0,2	0,1	
6	2	0,2	0,8	0,1	

Four following tables (Table 7, 8, 9, and 10) present results obtained by common classifier based on fusion of discriminate function according to four presented models respectively.

Table 7. Results for classifier that fuses discriminate function according to model 1.

		Elementary classifiers									Common classifier				Oracle
		classifier 1			classifier 2			classifier 3							
object	correct class	class 1 support	class 2 support	weight	class 1 support	class 2 support	Weight	class 1 support	class 2 support	weight	class 1 support	class 2 support	decision	fails	fails
1	1	0,68	0,32	0,2	0,51	0,49	0,8	0,98	0,02	0,1	0,64	0,32	1	0	0
2	1	0,45	0,55	0,2	0,12	0,88	0,8	0,96	0,04	0,1	0,28	0,54	2	1	0
3	2	0,48	0,52	0,2	0,99	0,01	0,8	0,47	0,53	0,1	0,94	0,91	1	1	0
4	1	0,49	0,51	0,2	0,87	0,13	0,8	0,48	0,52	0,1	0,84	0,86	2	1	1
5	1	0,47	0,53	0,2	0,99	0,01	0,8	0,47	0,53	0,1	0,93	0,92	1	0	1
6	2	0,05	0,95	0,2	0,58	0,42	0,8	0,15	0,85	0,1	0,49	1,22	2	0	0
													failures total	3	2

Table 8. Results for classifier that fuses discriminate function according to model 2.

		Elementary classifiers									Common classifier				Oracle
		classifier 1			classifier 2			classifier 3							
object	correct class	class 1 support	class 2 support	weight	class 1 support	class 2 support	weight	class 1 support	class 2 support	weight	class 1 support	class 2 support	decision	Fails	fails
1	1	0,68	0,32	0,2	0,51	0,49	0,2	0,98	0,02	0,1	0,34	0,12	1	0	0
2	1	0,45	0,55	0,3	0,12	0,88	0,2	0,96	0,04	0,2	0,31	0,29	1	0	0
3	2	0,48	0,52	0,3	0,99	0,01	0,2	0,47	0,53	0,1	0,39	0,31	1	1	0
4	1	0,49	0,51	0,3	0,87	0,13	0,2	0,48	0,52	0,1	0,37	0,31	1	0	1
5	1	0,47	0,53	0,3	0,99	0,01	0,2	0,47	0,53	0,1	0,39	0,31	1	0	1
6	2	0,05	0,95	0,3	0,58	0,42	0,2	0,15	0,85	0,1	0,15	0,36	2	0	0
													failures total	1	2

Table 9. Results for classifier that fuses discriminate function according to model 3.

		Elementary classifiers									Common classifier				Oracle
		classifier 1			classifier 2			classifier 3							
object	correct class	class 1 support	class 2 support	weight	class 1 support	class 2 support	weight	class 1 support	class 2 support	weight	class 1 support	class 2 support	decision	Fails	fails
1	1	0,68	0,32	0,2	0,51	0,49	0,8	0,98	0,02	0,6	1,13	0,57	1	0	0
2	1	0,45	0,55	0,2	0,12	0,88	0,8	0,96	0,04	0,6	0,76	1,00	2	1	0
3	2	0,48	0,52	0,2	0,99	0,01	0,8	0,47	0,53	0,6	1,17	1,04	1	1	0
4	1	0,49	0,51	0,8	0,87	0,13	0,2	0,48	0,52	0,1	0,61	0,43	1	0	1
5	1	0,47	0,53	0,8	0,99	0,01	0,2	0,47	0,53	0,1	0,62	0,44	1	0	1
6	2	0,05	0,95	0,8	0,58	0,42	0,2	0,15	0,85	0,1	0,17	0,38	2	0	0
													failures total	2	2

Table 10. Results for classifier that fuses discriminate function according to model 4.

		Elementary classifiers									Common classifier				Oracle
		classifier 1			classifier 2			classifier 3							
object	correct class	class 1 support	class 2 support	weight	class 1 support	class 2 support	weight	class 1 support	class 2 support	weight	class 1 support	class 2 support	decision	Fails	fails
1	1	0,68	0,32	0,2	0,51	0,49	0,2	0,98	0,02	0,6	0,83	0,37	1	0	0
2	1	0,45	0,55	0,2	0,12	0,88	0,2	0,96	0,04	0,6	0,69	0,67	1	0	0
3	2	0,48	0,52	0,2	0,99	0,01	0,2	0,47	0,53	0,6	0,58	0,42	1	1	0
4	1	0,49	0,51	0,2	0,87	0,13	0,2	0,48	0,52	0,6	0,56	0,47	1	0	1
5	1	0,47	0,53	0,2	0,99	0,01	0,1	0,47	0,53	0,6	0,48	0,31	1	0	1
6	2	0,05	0,95	0,2	0,58	0,42	0,8	0,15	0,85	0,1	0,49	1,22	2	0	0
													failures total	1	2

Comparison of the results obtained for all common classifier shows similar relation that was observed for model that fuse elementary classifier response. The more elements affect the weights the more flexible fuser is and the less misclassifications are made. In addition we can observe that fusing discriminating function brings more information about the object being under recognition as the total results of such fusers are slightly better comparing to fuser of classifier response. The most promising observation can be made in Table 10. We can see that common classifier that uses weights that depends on classifier, class number, and feature vector outperforms *Oracle*. Of course presented example shows only possibility of producing combining classifiers better than Oracle but we still do not know how to learn the fuser. Therefore we decide to carried out some computer experiments which confirm our propositions.

3. EXPERIMENTAL INVESTIGATION

The aim of the experiment is to compare the performance of fuser of class labels based on weights depended on classifier and class number realized as neural networks and genetic algorithm with Oracle classifier.

3.1. Experimental investigation

We used twelve databases from UCI Machine Learning Repository [1]: “Contracaptive”, “Connectionist”, “Letter”, “Glass”, “Ionosphere”, “Image”, “Hayes”, “Haberman”, “Highleyman”, “Hill”, “Difficult” and “Dermatology” which are described in Table 11.

Table 11. Databases' description.

No	Database	Number of	
		Classes	examples
1	Letter	3	20000
2	Glass	7	214
3	Ionosphere	2	351
4	Image	7	2310
5	Connectionist	11	528
6	Hill	101	606
7	Highleyman	2	500
8	Hayes	5	160
9	Haberman	2	306
10	Difficult	2	500
11	Dermatology	6	366
12	Contracaptive	2	1473

The set of five elementary classifiers consisting of slightly undertrained networks (for which training process were early stopped) has been used in experiments for the purpose of ensuring diversity of simple classifiers that allows their local competences to be exploited. The details of used neural nets are as follow:

- sigmoidal transfer function,
- back propagation learning algorithm,
- 5 neurons in hidden layer,
- number of neurons in last layer equals number of classes of given experiment.

They were used the following compound classifiers:

- NN which was realized as neural networks.
- AE which was realized as genetic algorithm.

Qualities of mentioned classifiers were compared with classical voting fuser of classifier response without any weighting (denoted as MV) and *Oracle* classifier (ORACLE).

It is also worth to notice that BP means the best classifier in pool of classifiers.

Other set-up of experiments were as follow:

1. All experiments were carried out in Matlab environment using the PRtools toolbox [8] and own software.
2. Classifiers' errors were estimated using the 10 fold cross validation method [16].

The results of experiments are presented in table 12 for each tested database respectively. For each database the experiment was repeated 10 times. The second and third bars that represent results obtained by NN and AE classifier respectively show average misclassification rate. No deviations for *Oracle* and Majority Voting were observed that classifiers (fusers) do not require training process and therefore their results are fixed and depend on elementary classifiers only.

Table 11. Results of experiments.





The results were also compared using 10-Fold Cross-Validated Paired t Test. Those results are presented in Table 12. Bolded results present data which are statistically different.

Table 12. Statistical comparison of experimental results for 12 databases using 10-Fold Cross-Validated Paired t Test

Database	Comparison of		
	Oracle with NN	Oracle with AE	NN with AG
Ionosphere	2,90	-1,60	-2,82
Letter	-2,21	-1,58	2,02
Image	2,79	0,12	-1,77
Hill	-2,24	-2,1	1,79
Highleyman	-2,26	-2,03	2,28
Hayes	1,76	-1,78	-2,39
Haberman	-1,88	-1,84	-1,24
Glass	0,16	-0,16	-0,54
Difficult	-2,14	-2,17	-0,61
Dermatology	2,23	2,29	0,24
Contracaptive	1,99	-1,80	-1,16
Connectionist	-0,59	-2,22	-2,19

3.2. Experimental results evaluation

Firstly, one has to note that we are aware of the fact that the scope of computer experiments was limited. Therefore, making general conclusions based on them is very risky.

In the case of the presented experiment:

1. Both tested classifiers that incorporate training and fuse elementary classifier with means of set of weighs, NN and AE, gained noticeable smaller misclassification rate then classifier making decision according simple majority voting algorithm MV and the best classifier in a pool BP. The only exception is result of NN for Highleyman and Ionosphere database. That is expected observation which proves that weighting significantly improve a quality of a resulting classifier.
2. In seven of twelve databases NN or AE classifier averagely obtained better result that Oracle. That is the most prominent result of our experiments that proves our statement that it is possible to outperform Oracle.

3. FINAL REMARKS

Chosen methods of classifier fusion were discussed in this paper and two of them were evaluated via computer experiments on benchmark and computer generated databases.

Obtained results justify the use of weighted combination and they are similar as published in [6, 17, 25]. Unfortunately, as it was stated, it is not possible to determine weight values in the analytical way. However, it is hoped that in practical situations the weights can be appropriately set, either with the aid of a suitable expert, or else by applying the data training methods proposed earlier in the paper.

ACKNOWLEDGEMENTS

The project co-financed by the European Union under the European Social Fund

REFERENCE

- [1] Asuncion, A., Newman, D.J., (2007), UCI Machine Learning Repository [<http://www.ics.uci.edu/~mllearn/MLRepository.html>]. Irvine, CA: University of California, School of Information and Computer Science.
- [2] Buckinx, W. et al, (2002), Using Machine Learning Techniques to Predict Defection of Top Clients, Proc. 3rd International Conference on Data Mining Methods and Databases, Bologna, Italy, pp 509–517.
- [3] Chow C.K., (1965), Statistical independence and threshold functions, *IEEE Trans. on Electronic Computers*, EC-16, pp 66-68.
- [4] Crook J.N., Edelman D.B., and Thomas L.C., (2007), Recent developments in consumer credit risk assessment, *European Journal of Operational Research*, 183, pp 1447-1465.
- [5] Duda R.O., et al., (2001), *Pattern Classification*, Wiley-Interscience.
- [6] Duin R.P.W., Tax, D.M.J., (2000), Experiments with Classifier Combining Rules, *LNCS*, No. 1857, pp 16–29.
- [7] Duin R. P.W.(2002), The Combining Classifier: to Train or Not to Train?, *Proc. of the ICPR2002*, Quebec City.
- [8] Duin R.P.W.et al. (2004), *PRTools4, A Matlab Toolbox for Pattern Recognition*, Delft University of Technology, 2004.
- [9] Fumera G., Roli F. (2005), A Theoretical and Experimental Analysis of Linear Combiners for
- [10] Hansen L.K., Salamon P. (1990), Neural Networks Ensembles, *IEEE Trans. on PAMI*, vol. 12, no. 10, pp 993-1001.
- [11] Hashem S. (1997), Optimal linear combinations of neural networks, *Neural Networks*, 10(4), pp 599-614.
- [12] Inoue H., Narihisa H. (2002), Optimizing a Multiple Classifier Systems, *LNCS*, Vol. 2417, pp 285–294.
- [13] Jozefczyk J. (1986), Determination of optimal recognition algorithm in two-level systems, *pattern recognition Letters* 4, pp 413-420.
- [14] Jain A.K., Duin P.W., Mao J.(2000), Statistical Pattern Recognition: A Review, *IEEE Trans. on PAMI*, vol 22., No. 1, pp 4-37.
- [15] Jackobs R.A., (1995) Methods for combining experts' probability assessment, *Neural Computation*, No. 7, pp 867–888.
- [16] Kohavi R. (1995), A study of cross-validation and bootstrap for accuracy estimation and model selection, *Proc. of the 14th Int. Joint Conf. on Artificial Intell.*, San Mateo, pp 1137–1143.

- [17] Kuncheva L.I., Bezdek J.C., Duin R.P.W.(2001), Decision templates for multiple classifier fusion: an experimental comparison, *Pattern Recognition*, 34, pp 299–314.
- [18] Kuncheva L.I., Whitaker C.J., Shipp C.A., Duin R.P.W. (2003), Limits on the Majority Vote Accuracy in Classifier Fusion, *Pattern Analysis and Applications*, 6, pp 22-31.
- [19] Kuncheva L.I. (2004), *Combining pattern classifiers: Methods and algorithms*, Wiley.
- [20] Raudys S. (2006), Trainable fusion rules. I. Large sample size case, *Neural Networks* 19, pp 1506-1516.
- [21] Raudys S. (2006), Trainable fusion rules. II. Small sample-size effects , *Neural Networks* 19, pp 1517-1527.
- [22] Shen A., Tong R., and Deng Y., Application of Classification Models on Credit Card Fraud Detection, Proc. of 2007 International Conference on Service Systems and Service Management, Chengdu, China, 9-11 June 2007, pp 1 – 4.
- [23] Tang T.-I et al., (2005), A Comparative Study of Medical Data Classification Methods Based on Decision Tree and System Reconstruction Analysis , *Industrial Engineering and Management Systems*, Vol. 4, No. 1, pp 102-108.
- [24] Tumer, K., Ghosh, J. (1996), Analysis of Decision Boundaries in Linearly Combined Neural Classifiers, *Pattern Recognition*, 29, pp 341–348.
- [25] Van Erp M., Vuurpijl L.G., Schomaker L.R.B. (2002), An overview and comparison of voting methods for pattern recognition, *Proc. of IWFHR.8*, Canada, pp 195–200.
- [26] Kuncheva L. (1995), Using degree of consensus in two-level fuzzy pattern recognition, *European Journal of Operational Research*, 80, pp 365-370.
- [27] Wozniak M. (2008), Experiments on linear combiners, (in:) Pietka E, Kawa J. (eds) *Information technologies in biomedicine*, Springer, pp 445-452.
- [28] Xu L., Krzyzak A., Suen Ch.Y.(1992), Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition, *IEEE Trans. on SMC*, no. 3, pp 418-435.