Combining One-Class Classifiers to Classify Missing Data

Piotr Juszczak and Robert P.W. Duin

Information and Communication Theory Group
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, The Netherlands
{p.juszczak,r.p.w.duin}@ewi.tudelft.nl

Abstract. In the paper a new method for handling with missing features values in classification is presented. The presented idea is to form an ensemble of one-class classifiers trained on each feature, preselected group of features or to compute from features a dissimilarity representation. Thus when any feature values are missing for a data point to be labeled, the ensemble can still make a reasonable decision based on the remaining classifiers. With the comparison to standard algorithms that handle with the missing features problem it is possible to build an ensemble that can classify test objects with all possible occurrence of missing features without retrain a classifier for each combination of missing features. Additionally, to train such an ensemble a training set does not need to be uncorrupted. The performance of the proposed ensemble is compared with standard methods use with missing features values problem on several UCI datasets.

1 Introduction

The increasing resolution of the sensors increases also the probability that one or a group of features can be missing or strongly contaminated by noise. Data may contain missing features due to a number of reasons e.g. data collection procedure may be imperfect, a sensor gathering information may be distorted by unmeasurable effects yielding the loss of data. Several ways of dealing with missing feature values have been proposed. The most simple and straightforward is to ignore all missing features and use the remaining observations to design a classifier [1]. Other group of methods estimates values of the missing features from available data by: replacing missing feature values by e.g. their means estimated on a training set [2], [3]. Morin [4] proposed to replace missing feature values by values of these features from their nearest neighbors, in the available, lower dimensional space, from the training set. [5, 4, 1] described different solutions using the linear regression to estimate substitutes for missing features values. However, Little [5] showed that many such methods are inconsistent, i.e. discriminant functions designed form the completed training set do not converge to the optimal limit discriminant functions as sample size tends to infinity. At this moment, methods recommended as generally best are based on EM algorithm [6, 1].

F. Roli, J. Kittler, and T. Windeatt (Eds.): MCS 2004, LNCS 3077, pp. 92–101, 2004.

However, because of the complexity of existing methods, neither of them can provides a solution for all cases with missing features, that can occur during classification. When an corrupted test point occurs a classifier is retrained or missing features is replaced by estimated ones which can lead to worse results than when the classification decision is based just on existing features [5, 1]. Additionally, in most of the proposed solutions to the missing feature problem it is assumed that training data is uncorrupted, thus potentially valuable data are neglected during training.

In this paper several techniques based on combining one-class classifiers [7] are introduced to handle missing feature. The classifiers are trained on one-dimensional problems, n-dimensional problem or features are combined in dissimilarity representations. The presented method can coupe with all possible situation of missing data, from single one to $N-1$, without retraining a classifier. It also makes use of corrupted data available for training.

The layout of this paper is as follows: in section 2, the problem of missing feature values and combining one-class classifiers (*occs*) is addressed. In section 2.1 some possibility of combining one-class classifiers to handle missing features problem are discussed. Section 3 shows results on UCI datasets and discusses the relative merits and disadvantages of combining *occs*. Section 4 presents the discussion and conclusions.

2 Formal Framework

Suppose two sets of data are given: a training set

$$\mathcal{L} = \{(\mathbf{x}_m, \mathbf{y}_m) \ : \ \mathbf{x}_m \in \mathbb{R}^{\mathbf{p}_m}; \quad m = 1 \ldots M\},$$

and a test set

$$\mathcal{T} = \{\mathbf{x}_t \ : \ \mathbf{x}_t \in \mathbb{R}^{\mathbf{q}_t}; \quad t = 1 \ldots T\}, \qquad \text{where} \quad (\mathbf{p}_m, \mathbf{q}_t) \in \mathbb{R}^N.$$

Where **x**-s represent objects and **y**-s represent labels[1]. N is a number of all the features considered in a classification task. Each object **x** in $\mathcal{L}$ or $\mathcal{T}$ can reside in the different space. Even if x_{m_1} and x_{m_2} are represented in spaces of the same dimensionality $\|p_{m_1}\| = \|p_{m_2}\|$, the present features might be different $p_{m_1} \neq p_{m_2}$. Such a problem is called a classification with missing data in training and test sets.

Suppose a classifier is designed by using uncorrupted data. Assume that input (test) data are then corrupted in particularly known ways. How to classify such corrupted inputs to obtain the minimal error? For example, consider a classifier for data with two features, such that one of the features is missing for a particular object x to be classified. Fig. 1 illustrates a three-class problem, where for the test object x the feature f_1 is missing. The measured value of f_2 for x is x_{f_2}. Clearly, if we assume that the missing value can be substituted by the $mean(f_1)$,

[1] It is assumed that both the training set $\mathcal{L}$ and the test set $\mathcal{T}$ are corrupted.

x will be classified as y_2. However, if the priors are equal, y_3 would be a better decision, because $p(x_{f_1}|y_3)$, estimated on the training set is the largest of the three likelihoods. In terms of a set of existing features $\mathbf{F}$, the posteriors are [8]:

$$P(y_i|\mathbf{F}) = \frac{\int g_i(\mathbf{F})\, p(\mathbf{F})\, d\mathbf{f}_{-}}{\int p(\mathbf{F})\, d\mathbf{f}_{-}} \tag{1}$$

where $\mathbf{f}_{-}$ indicates the missing features, $g_i(\mathbf{F}) = P(y_i|\mathbf{F}, \mathbf{f}_{-})$ is the conditional probability from a classifier. In short, equation (1) presents integrated, marginalization of the posterior probability over the missing features.

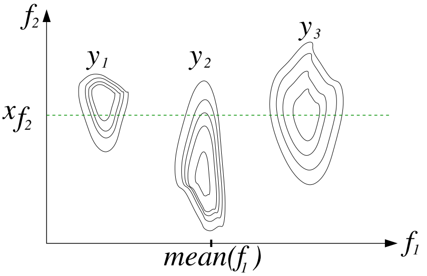

Fig. 1. Class conditional distributions for a three-class problem. If a test point misses the feature value from f_1 the optimal classification decision will be y_3 because $p(x_{f_1}|y_3)$ (estimated on the training set) is the largest.

Several attempts were made to estimate missing feature values for a test object [1] e.g. by:
- solving a classification problem in an available lower-dimensional feature space $\mathbf{F}$ (obviously in this case no estimation of the missing data is required);
- replacing missing feature values in $\mathcal{T}$ by the means of known values from $\mathcal{L}$;
- replacing missing values in $\mathcal{T}$ by values from the nearest neighbor from $\mathcal{L}$;
- using the expectation-maximization algorithm to maximize e.g. the class posteriors. This method is the most complicated one and the assumption about underlaying data distribution has to be made.

In further experiments the first and the third methods mentioned above are used as a comparison to the proposed methods.

2.1 Combining One-Class Classifiers

In the problem of one-class classification the goal is to accurately describe one class of objects, called the target class, as opposed to a wide range of other objects which are not of interest, called outliers. Many standard pattern recognition methods are not well equipped to handle this type of problem; they require complete descriptions for both classes. Especially when one class is very diverse and ill-sampled, usually (two-class) classifiers yield a very bad generalization for this class. Various methods have been developed to make such a data description [7]. In most cases, the probability density of the target set is modeled. This requires a large number of samples to overcome the curse of dimensionality [9].

Since during a training stage it is assumed that only target objects maybe present, a threshold is set on tails of the estimated probability or distance d such that a specified amount of the target data is rejected, e.g. 0.1. Then in the test stage, the estimated distances d can be transformed to resemble posterior probabilities as follow $p(y|x) = \frac{1}{1+e^{-d}}$ for the target class and $1 - p(y|x)$ for the outlier class.

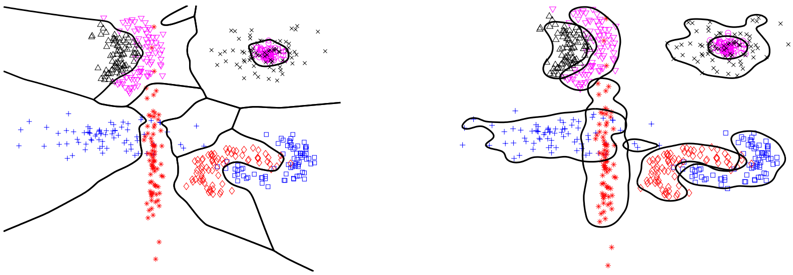

Fig. 2. A multi-class problem solved by: (left) combining two-class classifiers (one-vs-all approach), (right) combining one-class classifiers by the maximum resemblance to a model.

Fig. 2 illustrates differences between the solution to multi-class problem by combining two-class classifiers (one-vs-all approach) [9] and combining one class classifiers [7]. In the first approach, the entire data space is divided into parts being assigned to a particular class. A new object x **has to** be classified to one of the classes present in the training set. It means in a case of outliers the classification is ironies. In addition in one-vs-all or pairwise combining approach one has to compensate imbalance problem by e.g. settings probabilities to appropriate levels.

The right in Fig. 2 plot shows the *occs* combined by max rule. This means that in order to handle a multi-class problem, *occs* can be combined by the max rule or by a train combiner. In this approach, one assigns a new data point only to the particular class if it is in one of the described domains. If a new object x lies outside a region described by the target class, it is assigned to the outlier class. In the combination of two-class classifiers it appears that often the more robust mean combination rule is to be preferred. Here extreme posterior probability estimates are averaged out. In one-class classification only the target class is modeled $P(\mathbf{x}|\omega_{T_c})$ and a low uniform distribution is assumed for outlier class. This makes this classification problem asymmetric and extreme target class estimates are not canceled by extreme outlier estimates. However, the mean combination covers a broad domain in feature space [10], while the product rule has restricted range. Especially in high dimensional spaces this extra area will cover a large volume and potentially a large number of outliers.

2.2 Proposed Method

In this paper we propose several methods based on combing one-class classifiers to handle the missing features problem. Our goal is to build such an ensemble that dose not required retraining of a classifier for every combination of missing data and at the same time minimizes number of classifiers that has to be considered. In this section we will describe several ways of combining *occs* and some possibilities for the based classifiers.

First, two-class classifiers, combined like in one-vs-all method are considered; Fig. 2 (left) trained on all possible combination of missing feature values. In such case the number of base two-class classifiers that has to be trained is $K_{C,N} = (2^N - 1) \cdot \frac{C \cdot (C-1)}{2}$, where N is the number of features and C is the number of classes. Since all the features cannot be missing 1 is subtracted from all 2^N possibilities. For a problem with ten features and two classes, $K_{2,10} = 1023$ and for 20 features, $K_{2,20} = 1048575$. For such simple problems the number of classifier is already quite large.

On the other hand, if one-class classifiers are trained on all possible combination of missing features than the number of possibilities reduces to $K_{C,N} = (2^N - 1) \cdot C$ and the classification regions do not longer are considered as open spaces, Fig. 2 (right). However, for a large number of features this is a quite complicated study, since the number of classifiers is still cumbersome to handle and the system is difficult to validate.

In this paper, one of the proposed methods is to use one-class classifiers as base classifiers to combine, trained on one-dimensional problems and combine by fix combining rules: mean, product, max etc.. This reduces the number of classifiers that has to be in the pool as a combining possibilities to $N \cdot C$ for the fixed combining rules $K_{2,20} = 40$.

Below the way how to use fix (mean, product, and max) combining rules applied to the missing feature values problem in multi-class problems are described.

Mean combining rule: $y(\mathbf{x}|\omega_T) = \arg\max_c \left[\sum_{i=1}^{N'} P(x_i|\omega_{T_c}) \right]$

Product combining rule: $y(\mathbf{x}|\omega_T) = \arg\max_c \left[\prod_{i=1}^{N'} P(x_i|\omega_{T_c}) \right]$

Max combining rule: $y(\mathbf{x}|\omega_T) = \arg\max_c \left[\max_i P(x_i|\omega_{T_c}) \right]$

where $P(x_i|\omega_{T_c})$ is a probability that object x belongs to the target class C and N' is the number of available features. The probabilities $P(x_i|\omega_{T_c})$ estimated on single features are combined by fix rules. The test object $\mathbf{x}$ is classified to the class C with the maximum resemblance to it. However, because a single feature x_i is considered at time during classification the feature interactions are neglected. This can lower the performance of the proposed ensemble. This problem will be addressed in the section 3.2 of this paper.

Combining Dissimilarity Based Representations. The second method that is proposed in this paper, to handle missing features, is to combine non-missing features in the dissimilarity measure [11]. In this case, instead of training a classifier in the N' dimensional feature space it is trained in a dissimilarity space. In our experiments the sigmoid transformation from distances to dissimilarities is used:

$$dd_{jk} = \frac{1}{N'} \sum_{i=1}^{N'} \left[\frac{2}{1 + \exp(-\frac{d_{jk_i}}{\sigma_i})} - 1 \right] \qquad \text{where} \quad \sigma_i = \frac{\mathbf{d}_i}{N} \tag{2}$$

where dd_{jk} is the computed dissimilarity between object j and k and d_{jk_i} is the Euclidean distance between those two objects. To increase the robustness in the case of missing features dissimilarities were averaged out over one-dimensional representations. σ_i is approximated by the average distance between training objects considered the single feature i at time.

3 Experiments

In the experiments as the base classifier a simple Parzen classifier was used with the threshold set to reject 0.05 of target objects in the training set [7]. The smoothing parameter was optimized by leave-one-out approach [12]. Linear Programming Dissimilarity-data Description (LPDD) [13] (dist) was used as the base classifier for combining dissimilarity representations with 0.05 of the target objects rejection rate. To combine classifiers resemblances to the target class $y(x|\omega_T)$ were transformed to posterior probabilities by the sigmoid transformation $\frac{1}{1+\exp(-y(x|\omega_T))}$. The fixed (mean, product and max) [14] combining rules are applied to posterior probabilities computed from the resemblance on single features.

The proposed methods were compared to two standard methods designed to handle missing data: training classifier in a lower, available feature space (lower) and replacing missing features in a test object from $\mathcal{T}$ by the features of their nearest neighbor from a training set $\mathcal{L}$ (f_{nn}). The experiments were carried out on some of UCI datasets [15]: WBCD - Wisconsin Breast Cancer Dataset (number of classes c = 2, number of features k = 9), MFEAT (c=10, k=649), CBANDS (c=24, k=30), DERMATOLOGY (c=6, k=34), SATELLITE (c=6, k=36). The total number of features in MFEAT dataset was reduced from the original number of 649 to 100 (MFEAT 100) and 10 (MFEAT 10) by a forward feature selection based on maximization of the Fisher criterion: the trace of ratio of the within- and between-scatter matrices $J = tr\{S_W^{-1}S_B\}$ [16], to avoid the curse of dimensionality. The datasets were spited randomly into the equally sized training and test sets. For each percent of missing features ([0:10:90]%) ten training sets and for each training set ten test sets were randomly generated 10×10.

3.1 Combining *occs* Trained on Single Features

In this section the ensemble built from classifiers trained on individual features are evaluated. It is assumed that each feature contributes similar, independent amount of information to the classification problem. Any interactions between features are neglected.

In Fig. 3 mean errors for different solution to the missing features problem for different multi-class problem are presented. The classifiers are trained on one-dimensional problems and combined by fix combining rules. In dism method corespondent dissimilarities are computed and LPDD is trained on all one-class classification problems. The results are compared with two standard methods

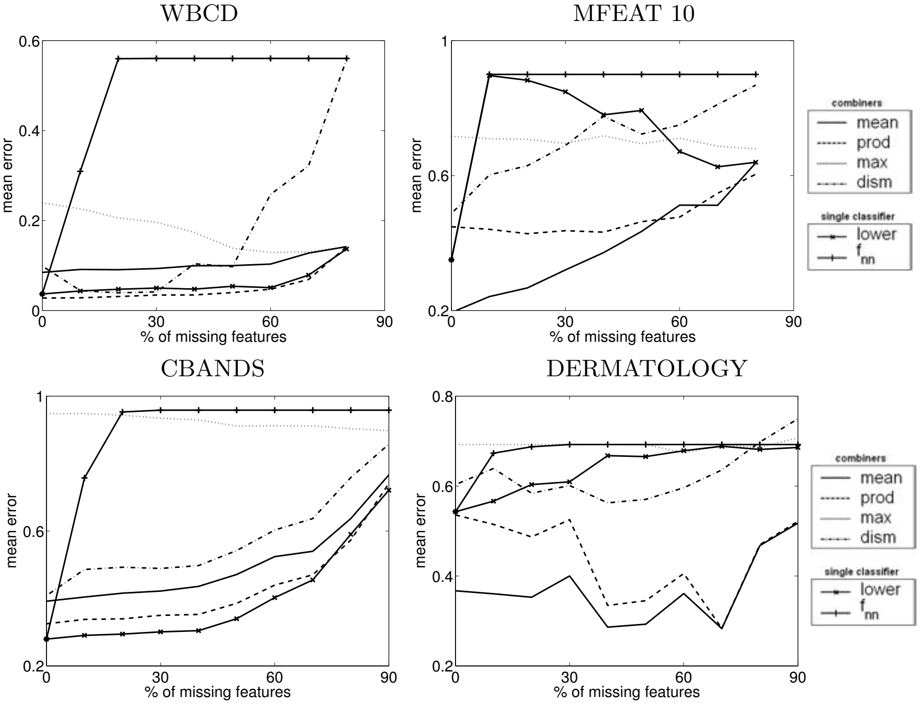

Fig. 3. Mean error for different percent of missing features for the combiners trained on single features for various combining rules: (mean, product, max), dism - dissimilarity representation LPDD. lower - the Parzen classifier trained on all available features. f_{nn} - the Parzen classifier trained on available features plus features from nearest neighbor from a training set. The results are averaged over 10×10 times; see text for details.

for missing features problem: lower - a classifier is trained on all available features neglecting missing features and f_{nn} missing feature values are replaced by features from the nearest neighbor of the test object in the training set. It can be observed that mean and product rule are performing the best for the entire range of missing features. It depends on the dataset which of this fix combining rules is better. The dissimilarity representation does not perform well, apart from WBCD, for which for a small percent of missing features the performance is comparable with fix combiners. The reason is that the computed dissimilarities on the training set, on all the features, are not resemble to dissimilarities computed on the test set with missing features. However, the dism method outperforms the standard f_{nn} method. The reasons for such poor performance of the f_{nn} method is that if more features are missing replacing them by features from the training set will cause less differences between test objects. The single classifier trained on all available features performs the best on the CBANDS dataset however is outperformed in other problems by fix combining rules. It can be concluded that more complicated problems split in simple ones and then combine can outperform a single, big classifier [17, 18].

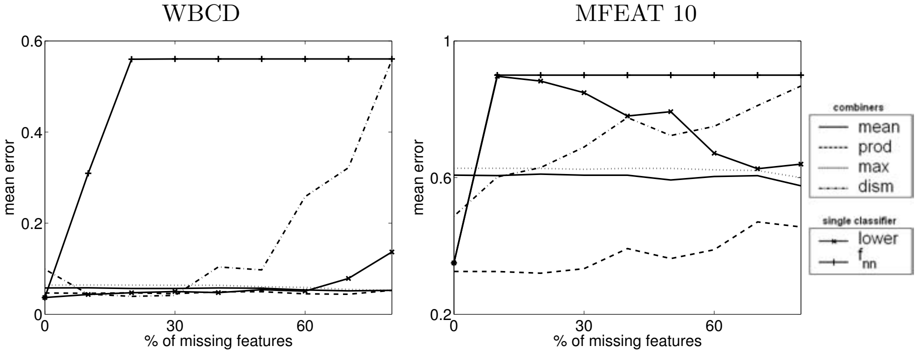

Fig. 4. Mean error for different percent of missing features for the combiners trained on (n+1) features for various combining rules: (mean, product, max), dism - dissimilarity representation LPDD. lower - the Parzen classifier trained on all available features. f_{nn} - the Parzen classifier trained on available features plus features from nearest neighbor from a training set. The results are averaged over 10 × 10 times; see text for details.

3.2 Combining *occs* Trained on $(n+1)$ Features

In the previous section it was assumed that every feature contributes a similar, independent amount of information to the classification problem. In this section we will study a possibility when a fixed number of features is always present or when there is a certain subset of features without which the classification is almost random e.g. for medical data like: name, age, height,..., examinations. It is probably possible to classify a patient to the healthy/unhealthy group without a name or age provided, but not without specific examination measurements. One of the possible solutions is to use a weighted combining rule [19].

In this paper, a different approach is proposed. Let us assume that the same n features are always present for the test objects. Therefore, instead of N possible missing features we have $N - n$ possibilities. In this case, we propose to train $(N - n - 1)$ base one-class classifiers in a $(n + 1)$-dimensional space. As a result, the base classifiers are highly depend. According to common knowledge on combining classifiers [14, 20], combining is beneficial when base classifiers differ. However, in our case, there is a trade-off between the number n of common features and how well the posterior probabilities are estimated. In Fig. 4, the mean error for $n = 3$ for WBCD and MFEAT 10 is shown. The standard deviation varies between 1-2% from the mean value. The classifiers are trained on (n+1) features and then combined. Compared to the results showed in Fig. 3 the performance of fix combiners increases. The posterior probabilities are better estimated and some features dependencies are also included in the estimation. If an additional knowledge is available about a classification problem e.g. n features are always present by appropriate combining better classification performance can be achieved.

3.3 Small Sample Size Problem

In this section the performance of the proposed method are evaluated for small sample size problems [9]. In small sample size problems the number of objects per class is similar or smaller than the number of features.

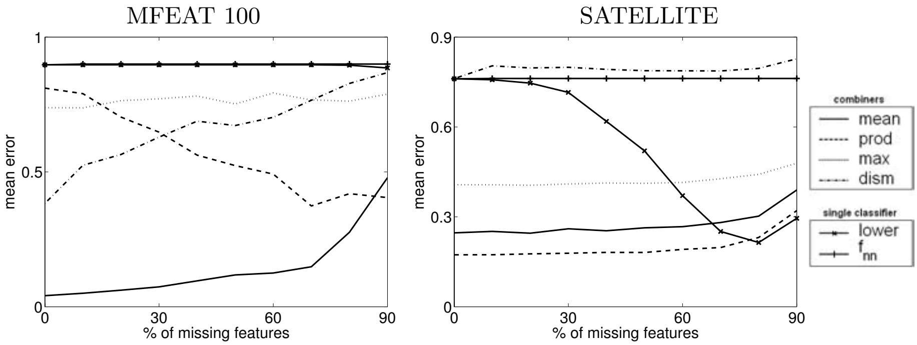

Fig. 5. Small sample size problems. Mean error for different percent of missing features for the combiners trained on single features for various combining rules: (mean, product, max), dism - dissimilarity representation LPDD. lower - the Parzen classifier trained on all available features. f$_{nn}$ - the Parzen classifier trained on available features plus features from nearest neighbor from a training set. The results are averaged over 10×10 times; see text for details.

Fig. 5 shows the mean error for two small sample size problems. Because the probabilities are estimated on single feature the proposed method is robust to small sample size problems. The classifier statistics are better estimated and the constructed ensemble is robust against noise.

4 Conclusions

In this paper, several methods for handling missing feature values have been proposed. The presented methods are based on combining one-class classifiers trained on one-dimensional or (n+1) dimensional problems. Additionally, the dissimilarity based method is proposed to handle the missing features problem. Compared to the standard methods, our methods are much more flexible, since they require much less classifiers to consider and do not require to retrain the system for each new situation when missing feature values occur. Additionally, our method is robust to small sample size problems due to splitting the classification problem to N several smaller ones.

Acknowledgments

Authors thank Elżbieta Pękalska for helpful suggestions in preparation the final version of the paper. This work was partly supported by the Dutch Organization for Scientific Research (NWO).

References

1. Little, R.J.A., Rubin, D.B.: Statistical analysis with missing data. 2 edn. ISBN 0-471-18386-5. Wiley-Interscience (2002)
2. Chan, L.S., Dun, O.J.: Alternative approaches to missing values in discriminant analysis. J. Amer. Statist. Assoc. **71** (1976) 842–844
3. Dixon, J.K.: Pattern recognition with partly missing data. IEEE Transactions on Sys., Man and Cyber. (1979) 617–621
4. Morin, R.L., Raeside, D.E.: A reappraisal of distance-weighted k-nearest neighbor classification for pattern recognition with missing data. IEEE Trans. Syst. Man Cybern. **11** (1981) 241–243
5. Little, R.J.A.: Consistent regression methods for discriminant analysis with incomplete data. J. Amer. Statist. Assoc. **73** (1978) 319–322
6. Ghahramani, Z., Jordan, M.I.: Supervised learning from incomplete data via an em approach. In: NIPS. (1994)
7. Tax, D.M.J.: One-class classification. PhD thesis, Delft University of Technology (2001)
8. Ahmad, S., Tresp, V.: Some solutions to the missing feature problem in vision. In: NIPS. (1993) 393–400
9. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern classification. 2nd edn. ISBN: 0-471-05669-3. Wiley Interscience (2001)
10. Tax, D.M.J., Duin, R.P.W.: Combining one-class classifiers. In: MCS. (2001) 299–308
11. Pekalska, E., Duin, R.P.W.: Dissimilarity representations allow for building good classifiers. PR Letters **23** (2002) 943–956
12. Duin, R.P.W.: On the choice of the smoothing parameters for parzen estimators of probability density functions. IEEE Transactions on Computers (1976)
13. Pekalska, E., Tax, D.M.J., Duin, R.P.W.: One-class lp classifiers for dissimilarity representations. In: NIPS. (2002) 761–768
14. Kittler, J., Hatef, M., Duin, R.P.W.: On combining classifiers. IEEE Transactions on PAMI **20** (1998)
15. Blake, C.L., Merz, C.J.: (UCI repository of machine learning databases)
16. Kittler, J.: Feature selection and extraction. Handbook of Pattern Recognition and Image Processing (1996) 59–83
17. Ho, T.K.: Data complexity analysis for classifier combination. In: MCS. (2001) 53–67
18. Raudys, S.: Multiple classification systems in the context of feature extraction and selection. In: MCS. (2002) 27–41
19. Littlestone, N., Warmuth, M.: Weighted majority algorithm. Information and Computation **108** (1994) 212–261
20. Duin, R.P.W.: The combining classifier: to train or not to train. In: ICPR. (2002)