Combining Fisher Linear Discriminants for Dissimilarity Representations

Elżbieta Pękalska, Marina Skurichina, and Robert P.W. Duin

Pattern Recognition Group, Department of Applied Physics,
Faculty of Applied Sciences, Delft University of Technology,
Lorentzweg 1, 2628 CJ Delft, The Netherlands

Abstract Investigating a data set of the critical size makes a classification task difficult. Studying dissimilarity data refers to such a problem, since the number of samples equals their dimensionality. In such a case, a simple classifier is expected to generalize better than the complex one. Earlier experiments [9,3] confirm that in fact linear decision rules perform reasonably well on dissimilarity representations.
For the Pseudo-Fisher linear discriminant the situation considered is the most inconvenient since the generalization error approaches its maximum when the size of a learning set equals the dimensionality [10]. However, some improvement is still possible. Combined classifiers may handle this problem better when a more powerful decision rule is found. In this paper, the usefulness of bagging and boosting of the Fisher linear discriminant for dissimilarity data is discussed and a new method based on random subspaces is proposed. This technique yields only a single linear pattern recognizer in the end and still significantly improves the accuracy.

1 Introduction

A difficult classification task arises when the training samples are far from being sufficient for representing the real distribution (the curse of dimensionality [8]). Simple decision rules, as linear classifiers, are expected to give lower generalization errors in such cases, since less parameters are to be estimated.

We are interested in applications in which the data is initially represented by a $n \times n$ dissimilarity matrix, e.g. all distances between a set of curves to be used for shape recognition. Our goal is to solve the recognition problem by a linear classifier, i.e. a linear combination of dissimilarities computed between the testing and training objects. In this representation the dimensionality k equals the number of samples: $k = n$ and one has to deal with the critical sample size problem. The Fisher linear discriminant (FLD) fails in such a case [9,3], since the estimated covariance matrix becomes singular.

The Pseudo-Fisher linear discriminant (PFLD) makes use of the pseudo-inverse, instead. However, $n = k$ reflects the worst situation for this classifier. It has been derived [10] and observed in reality [11] that the PFLD learning curve (generalization error as a function of training set size) is characterized by a peaking behavior exactly for this point (Figure 1), which is of our interest. However, some improvement is possible, either by using less objects or less features.

J. Kittler and F. Roli (Eds.): MCS 2000, LNCS 1857, pp. 117–126, 2000.

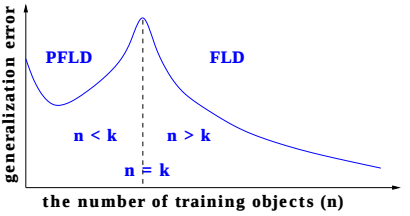


Figure 1. A typical learning curve for the (Pseudo)-Fisher linear discriminant.

Recently, the idea of combining (weak) classifiers has gained more attention. Combining simple pattern recognizers introduces some flexibility and can result in a more powerful decision rule in the end. A number of successful methods in this field exists. In this paper, we concentrate on boosting [4], bagging [1] and the random subspace method (RSM) [6,7] applied to dissimilarity data.

The paper is organized as follows. Section 2 gives some insight into dissimilarity-based pattern recognition. Boosting and bagging of the FLDs for distance data are discussed in section 3. A new technique operating in random subspaces is proposed in section 4. The simulation study on one artificial and two real datasets, alongside with the experimental set-up, is described in section 5. The results are discussed in section 6 and the conclusions are summarized in section 7.

2 The FLD for Dissimilarity-Based Pattern Recognition

In the traditional approach to learning from objects classifiers are constructed in a feature space. Dissimilarity-based pattern recognition offers alternative ways for building classifiers on dissimilarity (distance) representations. This can be especially of use, when the original data consist of a large set of attributes. In some cases it may be also easier or more natural to formulate a dissimilarity measure between objects than explicitly the features. Such measures differ according to various datasets or applications. For classification purposes, it is assumed that distances between two different objects are positive and zero otherwise.

A straightforward way of dealing with such a problem is based on relations between objects, which leads to the rank-based methods, e.g. the nearest neighbor rule. Another possibility is to treat distances as a description of a specific feature space, where each dimension corresponds to an object. This does not essentially change the classical feature-based approach, although a special case is considered: $n = k$ and each value expresses the magnitude of dissimilarity between two objects. In general, any arbitrary classifier operating on features can be used. In the learning process, the pattern recognizers are built on the $n \times n$ distance matrix. The p test objects are classified by using their distances to the n training samples (the test data consists of $p \times n$ dissimilarities).

Our earlier experiments [9] show that the feature-based classifiers operating on dissimilarity data often outperform the rank-based ones. Linear classifiers are of interest because of their simplicity. Distances are often built as a sum of many values and, under general conditions, they are approximately normally

distributed. Therefore, a normal-based classifier seems to be a reasonable one. Its simplest representative is the linear decision rule, assuming the same covariance matrix for all classes. For 2 equally probable classes it is given by [5]:

$$f(\boldsymbol{x}) \;=\; [\boldsymbol{x} - \frac{1}{2}(\overline{\boldsymbol{x}}_{(1)} + \overline{\boldsymbol{x}}_{(2)})]^T \, S^{-1} \, (\overline{\boldsymbol{x}}_{(1)} - \overline{\boldsymbol{x}}_{(2)}) \;=\; \boldsymbol{w}^T \, \boldsymbol{x} + w_0,$$

where: $\boldsymbol{w} \;=\; S^{-1} \, (\overline{\boldsymbol{x}}_{(1)} - \overline{\boldsymbol{x}}_{(2)})$ and $w_0 \;=\; -\frac{1}{2} \, (\overline{\boldsymbol{x}}_{(1)} + \overline{\boldsymbol{x}}_{(2)})^T \, \boldsymbol{w}$. This is equivalent to the FLD, obtained by maximizing the ratio of between-scatter to within-scatter (Fisher criterion [5]). Therefore, we refer to this function as to the FLD.

However, the rank r of the estimated covariance matrix S is smaller than n and its inverse cannot be found. The PFLD, using a pseudo-inverse operation, is proposed instead. The pseudo-inverse relies on the singular value decomposition of the matrix S and it becomes the inverse of S in the subspace spanned by the eigenvectors corresponding to r non-zero eigenvalues. The classifier is found in this subspace and to which it is orthogonal in the remaining $n - r$ directions.

A linear classifier can be also found by using the support vector approach [2]. However, in such a sparse distance space most of the objects become support vectors which means that the classifier is based on a high number of learning samples. This is not optimal, since the training relies on solving a difficult quadratic programming problem and the obtained result yields nearly no redundancy.

3 Boosting and Bagging for the PFLD

Boosting [4] is a method designed for combining weak classifiers, which are obtained sequentially during training by using the weighted objects. At each step the incorrectly classified objects from the previous step are emphasized with larger weights. Such (misclassified) samples tend to lie close to the class boundary, so they play a major role in building a classifier, indirectly approximating the support vectors [2]. However, when the learning set is not large enough, nearly all training objects are correctly classified. As a result, not much variation in weights is introduced, which makes all constructed classifiers alike. Consequently, very little can be gained by their combination [12]. Therefore, boosting seems not to be an appropriate method for our distance representations, where $n = k$.

Studying the PFLD learning curve (Figure 1) two possible approaches can improve the situation, when $n = k$. The first one tries to reduce the number of objects (going to the left side of the peak), while the second - the number of features (going to the right side of the peak, by shifting the curve to the left). The first idea can be put into practice by bagging and the second - by combing classifiers in random subspaces.

Bagging [1] is based on bootstrapping and aggregating, i.e. on generating multiple versions of a classifier and obtaining an aggregated (combined) decision rule. Using bootstrap replicates relates to unstable classifiers, for which a small change in the learning set causes a large change in their performance. Combining classifiers and emphasizing those which give better results, may finally lead to substantial gains in accuracy. Many rules exist for combining linear classifiers,

such as average (weighted) majority vote or by applying some operation (mean, product etc) on posterior probabilities of the combined classifiers.

Because of bootstrap characteristics, bagging may be of use in our case. In the training process the number of different objects is reduced, and we are practically placed in a situation in which dimensionality is larger then the number of samples (left side of the peak in Figure 1). This potentially enables us to construct a set of better performing classifiers and a more powerful decision rule in the end.

4 The FLD in Random Subspaces

It is known that multiple-tree and nearest neighbor classifiers combined in random subspaces [6,7] can gain a high accuracy. They outperform the single classifier constructed in the original space. The RSM, as an indeterministic approach, is based on a stochastic process in which a number of features is randomly selected. A classifier is then constructed in a subspace defined by those features. Proceeding in this way, a high-dimensional space can be exploited more effectively. The individual classifiers are built in subspaces, in which they are better defined. They are able to generalize well, although they do not have the full discrimination power. This stochastic process introduces some independence between classifiers and by combining them a better performance may be achieved.

This approach seems to be suitable for our problem, since it can profit from the high-dimensional data by exploring the possibilities in subspaces, thus it does not suffer from the curse of dimensionality [8]. Hopefully, the chosen dimensionality will turn out to be small so that the classifiers can be built in a cheap way. However, this issue has to be discussed and verified in practice. Another question refers to the number of subspaces needed to get a high accuracy.

Our proposal is to combine the FLDs in this stochastic way. Since the PFLD achieves its worst accuracy for $n = k$, the RSM may improve the performance in this case. The individual classifiers are built in subspaces of the fixed dimensionality and combined by averaging their coefficients, which yields only one linear classifier in the end. This is the advantage over combining rules based on posterior probabilities of the classifiers, where all of them should be stored for this purpose. Our RSM algorithm, called PF-RSM1, is briefly presented below:

K - the pre-defined number of selected features
for i=1 to N (the pre-defined number of combined classifiers) do
 Select randomly K features: $f_{i_{p(1)}}, \ldots, f_{i_{p(K)}}$;
 Build the FLD in a subspace obtaining the coef.: $w_{i_{p(1)}}, ..., w_{i_{p(K)}}, w_{i_0}$;
 Set to zero all coefficients of the ignored dimensions;
end
Determine the final decision rule with the coefficients: $w_1, ..., w_n, w_0$
by averaging the coefficients of all classifiers (including the
introduced zeros), i.e. $w_0 = \frac{1}{N} \sum_{i=1}^{N} w_{i_0}$ and $w_j = \frac{1}{N} \sum_{i=1}^{N} w_{i_j}, \ j = 1, ..., n$;

A slightly different version of this algorithm, namely PF-RSM2, is considered by using a validation set for the FLD trained in a subspace. This set is used to

determine a scaling factor for the FLD's coefficients. The scaling is done in such a way that the classified objects represent as well as possible posterior probabilities on the validation set. This does not influence the decision boundary itself.

In the proposed way of combining classifiers, although they are designed in subspaces, they are finally treated in the original space. This is achieved by setting the coefficients of the ignored dimensions to zero. Therefore, the final combination procedure (averaging) addresses them in the original, high-dimensional distance space. By doing this, the most preferable directions in the original space are emphasized and by including more and more classifiers all coefficients of the final decision rule become more accurate. It seems to be also possible to combine the classifiers explicitly in subspaces, which is an interesting concept for further research.

5 Datasets and Experiments

One artificial and two real datasets are used in our experimental study. The first set consists of 200-dimensional correlated Gaussian data [11]. There are two classes, each represented by 100 samples.

The second set is derived from NIST database [13] and consists of 2000 16×16 images of digits evenly distributed over 10 classes. In our simulations a 2-class problem was considered, for digits 3 and 5, to which we refer as to Digit35.

Vibration was measured with 5 sensors mounted on a submersible pump operating in one normal and 3 abnormal states [14]. The data consists of the wavelet decomposition of the power spectrum. For each sensor the 100 coefficients with the largest variances were considered. A 2-class problem was studied here to which we refer as to Pump2. It is described by 500 features and 450 samples equally distributed over 2 classes: bearing failure and loose foundation.

The squared Euclidean distance was considered for our experiments. For each dataset, the dissimilarity representation was computed, which became then our starting point for a recognition problem. Only the 2-class situations were investigated, since for binary problems the linear classifier is uniquely defined and our aim is to illustrate the potential of combination such simple pattern recognizers. This dissimilarity measure was chosen as an example, since our goal is not to optimize the classification error for the given data with respect to the distance measure used, but rather present what may be gained by combining single decisions for such problems.

Table 1. Characteristics of the datasets used in experiments.

	Gaussian	**Digit35**	**Pump2**
Original dimensionality	200	256	500
Number of samples for TR/TE	100 / 100	100 / 300	150 / 300
Distance representation for TR	100×100	100×100	150×150
Distance representation for TE (no valid. set)	100×100	300×100	300×150
Distance representation for TE (a valid. set)	66×100	266×100	250×150

A simulation study was done for boosting, bagging and the RSM. All the experiments were run 25 times. For the artificial dataset, 25 different sets were randomly drawn from the multi-normal distribution according to the specified parameters. For the real datasets, they were randomly split into the training and testing sets 25 times, each time taking care that prior probabilities for two classes remain equal. Table 1 shows characteristics of the explored sets.

6 Discussion

Boosting (see [12]) performs poorly on the investigated datasets. It does not improve accuracy of the single PFLD at all. In each run all objects are equally weighted, so the final decision rule is based on multiple identical discriminants. Therefore no boosting results are present in Figure 2.

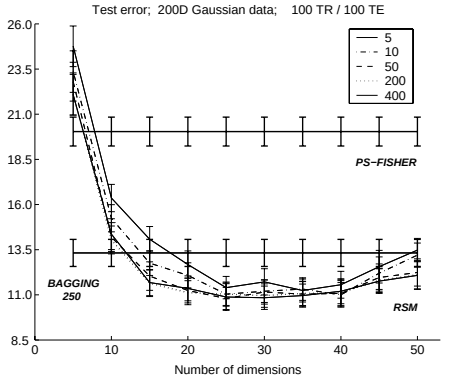
Boosting relies on the weighted majority vote, the RSM is based on the average, therefore the bagging experiment is conducted for both cases. In Figure 2, for clarity only, the error bar of bagging based on the average of 250 classifiers is plotted. For all datasets, the generalization errors reached by this combination rule and the weighted majority vote are very similar. The differences are however larger, when the number of combined classifiers is small, in disfavor of the weighted majority. Bagging seems to work well for the datasets under study; the accuracy is improved considerably by about $60\% - 65\%$, which is a beneficial achievement over the PFLD result. The details are shown in Table 2.

Table 2. The averaged generalization error and standard deviation (in %) for bagging.

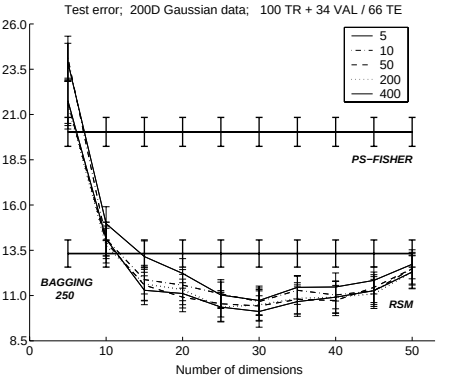
No. of	**Gaussian**		**Digit35**		**Pump**	
PFLD	Average	Majority	Average	Majority	Average	Majority
5	14.36 (0.67)	14.88 (0.63)	6.43 (0.29)	6.59 (0.25)	8.64 (0.34)	8.92 (0.40)
10	13.68 (0.69)	14.28 (0.66)	5.71 (0.26)	6.05 (0.26)	8.19 (0.38)	9.09 (0.38)
50	13.44 (0.73)	13.88 (0.77)	5.36 (0.24)	5.41 (0.24)	7.87 (0.39)	7.88 (0.38)
100	13.64 (0.74)	13.72 (0.77)	5.37 (0.24)	5.47 (0.24)	7.61 (0.38)	7.67 (0.34)
250	13.32 (0.75)	13.68 (0.80)	5.31 (0.21)	5.24 (0.20)	7.71 (0.37)	7.73 (0.37)

The RSM, as our proposal, is more thoroughly investigated. The dependency on the number of combined classifiers is studied and the dimensionality of the subspaces, as well. The results of our experiments are presented in Figure 2. The left/right pictures represent the situation either without or with a validation set. Its role is to scale the coefficients of the FLD found in a subspace so that the classified objects can represent as well as possible posterior probabilities. The number of samples used for a validation set was about 1/3 of the training set. It seems to be enough for determination of one scaling factor.

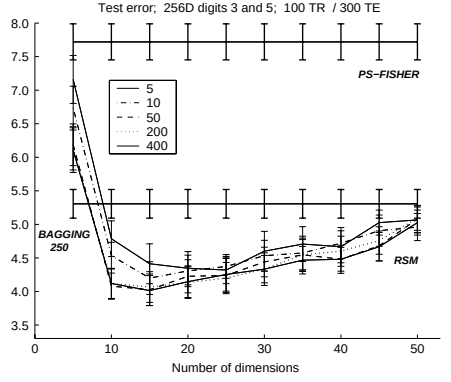
In our experiments, the RSM seems to work very well, accomplishing in its best case about $90\% - 110\%$ improvement over the PFLD result in the original space, competing the bagging achievements. The curves of the generalization error versus the subspace dimensionality indicate that in fact a small number of selected dimensions gives good results. This is essential, since in a low-dimensional



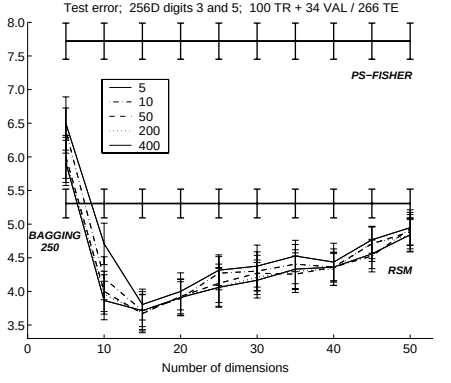
(a) Gaussian data; PF-RSM1.



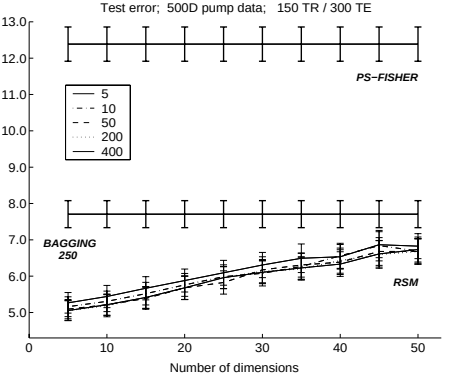
(b) Gaussian data; PF-RSM2.



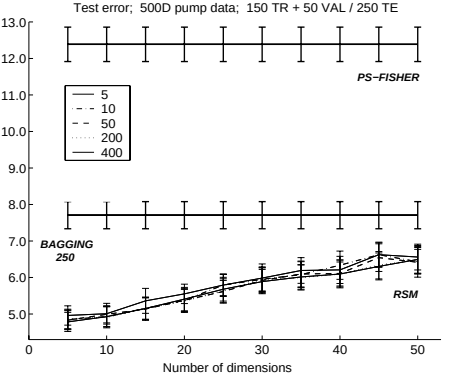
(c) Digit35 data; PF-RSM1.



(d) Digit35 data; PF-RSM2.



(e) Pump2 data; PF-RSM1.



(f) Pump2 data; PF-RSM2.

Figure 2. The generalization error (in %) of the PFLD compared to its bagging version and to the RSM. The legend refers to the number of the FLDs combined.

space, the FLD can be determined in a computationally cheaper way. The subspace dimensionality deviates around $7\% - 15\%$ of all features. The best result is observed for the Pump2 case (see Figure 2(e)-(f)), when the gain in accuracy is more than twice for 5 dimensions.

One notices also that already 5 or 10 combined FLDs decrease the generalization error substantially. Considering, e.g. the Digit35 data, it can be seen from Figure 2(c)-(d), that very small error is achieved in case of 5 combined classifiers for 15-dimensional subspaces. It gives us not more than 75 dimensions needed in total. For the Pump2 data (Figure 2(e)-(f)) this is even better, since the performance is improved already for 5-dimensional subspaces. So, the method makes use of not more than 25 features in the end. This is an important observation, suggesting that in practice a part of information may be skipped (especially of interest for large distance data), while gaining a high accuracy.

Table 3. The averaged error (in %) for the RSM with different combining rules.

No of. dim.	Average	Min	Mean	Median	Max	Product
	Gaussian					
5	22.08	18.04	22.60	25.12	18.04	22.08
10	14.36	13.12	14.44	14.56	13.12	14.36
15	11.68	12.36	11.64	11.80	12.36	11.68
20	11.36	11.40	11.56	11.36	11.40	11.36
35	10.96	12.28	10.84	10.96	12.28	11.08
50	12.08	15.36	11.92	11.94	15.36	14.60
	Digit35					
5	6.11	4.76	6.45	6.71	4.76	6.08
10	4.12	4.81	4.21	4.28	4.81	4.61
15	4.01	6.00	3.97	3.99	6.00	5.47
20	4.15	6.16	3.95	3.97	6.16	6.12
35	4.52	7.12	4.41	4.37	7.12	7.04
50	5.03	7.04	5.01	4.99	7.04	7.04
	Pump2					
5	5.05	5.29	5.05	5.00	5.29	5.05
10	5.21	5.48	5.21	5.16	5.48	5.21
15	5.41	5.72	5.43	5.41	5.72	5.41
20	5.68	6.00	5.65	5.61	6.00	5.69
35	6.23	6.64	6.21	6.17	6.64	6.53
50	6.70	8.60	6.64	6.60	8.60	8.37

With the growing number of subspace dimensions the generalization error first decreases, reaches its minimum and then starts to increase. The rule of thumb says that the classifiers generalize well when the number of training objects is e.g. 5-10 times larger than the number of features. Therefore, the increase of generalization error with the dimensionality larger then 5 (the Pump2 data), 15 (the Digit35 data) or $25 - 30$ (the Gaussian data) is not surprising. When the number of features k slowly approaches the number of objects n ($k \to n$), the FLD is going in the direction of the PFLD. It is then characterized by worse

performance (we are somewhat to the left of the peak in Figure 1), and their combination yields a worse decision rule, as well. Using a validation set seems to improve the results slightly, however one hoped that adjusting the FLD's coefficients would give much better improvement. It is possible, however, that another way of scaling may gain that.

One could argue whether other combing rules are not significantly better than our average-based RSM. Therefore, some of them were also studied. The comparison is presented in Table 3. As it can be noticed, the error obtained by average is very close to that one obtained by the mean rule and similar to that one gained by the median rule. This proves our point that averaging is useful, also especially it is computationally more efficient and yields only one classifier.

7 Conclusions

Studying distance representations may become useful when the data is characterized by many features or when experts cannot define the right attributes, but they are able to provide a dissimilarity measure, instead. The classical approach to such data is the rank-based one, namely the (condensed) nearest neighbor rule. We argue [3,9] that the feature-based approach, in which linear classifiers are built in the distance space can be more beneficial. However, in such a case one deals with the critical training size problem, since the number of training objects equals their dimensionality. Therefore, the usefulness of boosting, bagging and the RSM of the FLDs for dissimilarity representations has been investigated here. The novelty of our approach is that we concentrate on distance data, which is specific because of its $n \times n$ training size and because of its nature, i.e. relative information on objects and the structure being given in the data values.

It is also important to emphasize here that the combined classifiers can be advantageous when the generalization error of the single PFLD is higher than the overlap between classes. When it approaches the Bayes error, not much improvement may be gained. As an example, the squared Euclidean distance was studied as a measure of dissimilarity. Our goal is to investigate what may be achieved by combining single decisions for distance data. From our experiments the following conclusions can be drawn:

Firstly, boosting is not advantageous for our problem. It does not improve accuracy of the the single PFLD at all. No variation in weights is introduced during the training and the final decision rule is built from multiple identical discriminants. As suggested in [12], boosting is useful for large learning sizes.

Secondly, bagging, based either on the average or on the majority vote, improves the PFLD performance for all datasets studied. The achievement is about $60\% - 65\%$, which is a considerable value. By using bootstrap replicates, the number of different samples is reduced, so bagging deals practically with the situation when $n < k$. We are then placed on the left side of the peak of the PFLD learning curve (see Figure (1)).

Finally, we have proposed to combine the FLDs in random subspaces. Our technique yields a single linear classifier and gives the best improvement in

accuracy, which is about 100% for our datasets. The method constructs the FLDs in randomly selected subspaces of a fixed dimensionality and combines them by averaging their coefficients in the end. The experiments show that the best results are reached when a validation set was used and when the number of chosen dimensions deviates between 4% (The Pump2 data) and 30% (the Digit35 data) of all features. It allows for building classifiers in a cheap way. Even for a small number of combined classifiers, e.g. 5 or 10, the generalization error decreases substantially. This suggests that in practice some dimensions can be skipped, which is important for high-dimensional data.

8 Acknowledgments

This work was partly supported by the Foundation for Computer Science Research in The Netherlands (SION) and the Dutch Organization for Scientific Research (NWO).

References

1. L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.
2. C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.
3. R. P. W. Duin, E. Pękalska, and D. de Ridder. Relational discriminant analysis. *Pattern Recognition Letters*, 20(11-13):1175–1181, 1999.
4. Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Machine Learning: Proc. of the 13th International Conference*, pages 148–156, 1996.
5. K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Acad. Press, 1990.
6. T. K. Ho. Nearest neighbours in random subspaces. In *Proceedings of the Second International Workshop on Statistical Techniques in Pattern Recognition*, pages 640–648, Sydney (Australia), 1998.
7. T. K. Ho. The random subspace method for constructing decision forest. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998.
8. A. K. Jain and B. Chandrasekaran. Dimensionality and sample size considerations in pattern recognition practice. In P. R. Krishnaiah and L. N. Kanal, editors, *Handbook of Statistics*, volume 2, pages 835–855. North-Holland, Amsterdam, 1987.
9. E. Pękalska and R. P. W. Duin. Classifiers for dissimilarity-based pattern recognition. In *ICPR*, Barcelona (Spain), 2000, accepted.
10. S. Raudys and R. P. W. Duin. On expected classification error of the Fisher linear classifier with pseudo-inverse covariance matrix. *Pattern Recognition Letters*, 19(5-6), 1998.
11. M. Skurichina and R. P. W. Duin. Bagging for linear classifiers. *Pattern Recognition*, 31(7):909–930, 1998.
12. M. Skurichina and R. P. W. Duin. Boosting in linear discriminant analysis. In *First International Workshop on Multiple Classifier Systems*, Cagliari (Italy), 2000.
13. C.L. Wilson and M.D. Garris. Handprinted character database 3. Technical report, National Institute of Standards and Technology, February 1992.
14. A. Ypma, D. M. J. Tax, and R. P. W. Duin. Robust machine fault detection with independent component analysis and support vector data description. In *IEEE International Workshop on Neural Networks for Signal Processing*, pages 67–76, Wisconsin (USA), 1999.