Discriminant Projections Embedding for Nearest Neighbor Classification

Petia Radeva and Jordi Vitrià

Computer Vision Centre and Dept. Informàtica
Universitat Autònoma de Barcelona
08193 Bellaterra (Barcelona), Spain
{petia,jordi}@cvc.uab.es

Abstract. In this paper we introduce a new embedding technique to linearly project labeled data samples into a new space where the performance of a Nearest Neighbor classifier is improved. The approach is based on considering a large set of simple discriminant projections and finding the subset with higher classification performance. In order to implement the feature selection process we propose the use of the adaboost algorithm. The performance of this technique is tested in a multiclass classification problem related to the production of cork stoppers for wine bottles.

1 Introduction

One of the most common steps when designing a classifier system is to transform the original data representation to a new representation that is built by combining the original data features. This is called the feature extraction process. We can use different criteria to build this process. One of such criteria is the level of compactness that we get with the new input data representation, that leads to different dimensionality reduction techniques. In our case we focus in a different kind of criterium: discriminability. In this case the feature extraction process takes into account class membership of the input data to learn invariant data features that increase the classification ratios of the system.

Our objective is to find an embedding from the original data representation space to a new one that is specially designed to increase the performance of the nearest neighbor classification rule. We have not made assumptions on the data distribution, and we don't force our projection to be orthogonal [2]. The only assumption we impose is that our embedding must be based on a set of simple 1D projections, which can complement each other to achieve better classification results. We have made use of Adaboost algorithm [9] as a natural way to select feature extractors, and the coefficients that can rank the importance of each projection.

1.1 Discriminant Analysis

Discriminant analysis is a feature extraction tool based on a criterion J and two square matrices $\boldsymbol{S}_b$ and $\boldsymbol{S}_w$. These matrices generally represent the scatter of

A. Sanfeliu et al. (Eds.): CIARP 2004, LNCS 3287, pp. 312–319, 2004.

sample vectors between different classes for $\boldsymbol{S}_b$, and within a class for $\boldsymbol{S}_w$. The most frequently used criterion is to choose $J = trace(S_w^{-1} S_b)$.

It can be seen that, maximization of J is equivalent to finding the $D \times M$ linear transformation $\boldsymbol{W}$ such that

$$\hat{\boldsymbol{W}} = \arg \max_{\boldsymbol{W}^T \boldsymbol{S}_w \boldsymbol{W} = \boldsymbol{I}} trace(\boldsymbol{W}^T \boldsymbol{S}_b \boldsymbol{W}) \tag{1}$$

where $\boldsymbol{I}$ is the identity matrix. It can be proven that, given N samples of D dimensional data $\boldsymbol{X}$ and discriminant space dimensionality M, there is general method to solve the optimization problem given in equation (1) [5].

1.2 Fisher Discriminant Analysis

The most widely spread approach for discriminant analysis is the one that makes use of only up to second order statistics of the data. This was done in a classic paper by Fisher [1], and it is called Fisher Discriminant Analysis (FDA). In FDA the within class scatter matrix is usually computed as a weighted sum of the class-conditional sample covariance matrices where the weights are given by the class prior probabilities,

$$\boldsymbol{S}_w = \sum_{k=1}^{K} P(C^k) \boldsymbol{\Sigma}^k \tag{2}$$

where $\boldsymbol{\Sigma}^k$ is the class-conditional covariance matrix, estimated from the sample set. On the other side, the most common way of defining the between class-scatter matrix is as,

$$\boldsymbol{S}_b = \sum_{k=1}^{K} P(C^k)(\boldsymbol{\mu}^k - \boldsymbol{\mu}^0)(\boldsymbol{\mu}^k - \boldsymbol{\mu}^0)^T \tag{3}$$

where μ^k is the class-conditional sample mean and μ^0 is the unconditional (global) sample mean. Many other less spread out forms, always based on sample means and class-conditional covariance matrices are also available for these two scatter matrices [5]. The two main drawbacks of FLD are: Gaussian assumption over the class distribution of the data samples; and the dimensionality of the subspaces obtained is limited by the number of classes.

1.3 Nonparametric Discriminant Analysis

In [3] Fukunaga and Mantock present a linear and nonparametric method for discriminant analysis in an attempt to overcome the limitations present in (FDA) [1], and name the technique Nonparametric Discriminant Analysis (NDA).

In NDA we define a between-class matrix as the scatter matrix obtained from vectors locally pointing to another class. This is done as follows: Given a norm $\|\|$ in the metric space where the samples are defined, the extraclass nearest neighbor for a sample $\boldsymbol{x} \in C^k$ is defined as

$$\boldsymbol{x}^E = \{\boldsymbol{x'} \in \overline{C^k} / \|\boldsymbol{x'} - \boldsymbol{x}\| \leq \|\boldsymbol{z} - \boldsymbol{x}\|, \forall \boldsymbol{z} \in \overline{C^k}\} \tag{4}$$

where $\overline{C^k}$ notes the complement set of C^k. In the same fashion we can define the intraclass nearest neighbor as

$$\boldsymbol{x}^I = \{\boldsymbol{x'} \in C^k / \|\boldsymbol{x'} - \boldsymbol{x}\| \leq \|\boldsymbol{z} - \boldsymbol{x}\|, \forall \boldsymbol{z} \in C^k\} \tag{5}$$

Both definitions (4) and (5) can be extended to the K nearest neighbors case by defining $\boldsymbol{x}^E$ and $\boldsymbol{x}^I$ as the mean of the K nearest extra or intra-class samples. From these neighbors or neighbor averages, the extraclass differences are defined as $\boldsymbol{\Delta}^E = \boldsymbol{x} - \boldsymbol{x}^E$ and the intraclass differences as $\boldsymbol{\Delta}^I = \boldsymbol{x} - \boldsymbol{x}^I$. Notice that $\boldsymbol{\Delta}^E$ points locally to the nearest class (or classes) that does not contain the sample. The nonparametric between-class scatter matrix is then defined as

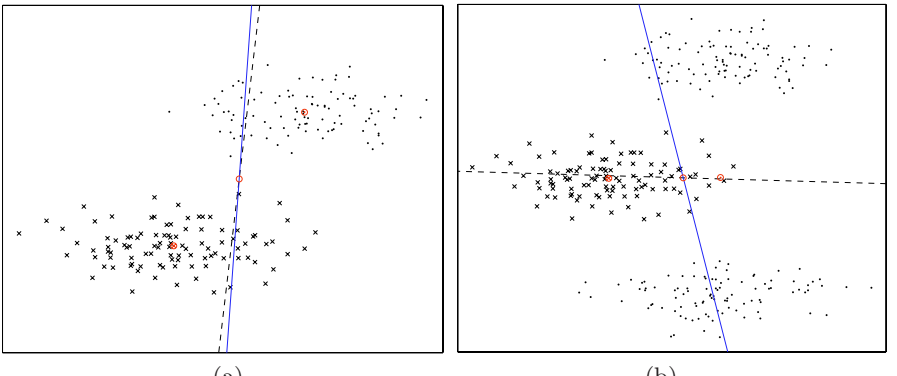
$$S_b = \frac{1}{N} \sum_{n=1}^{N} w_n (\boldsymbol{\Delta}_n^E)(\boldsymbol{\Delta}_n^E)^T \tag{6}$$

where $\boldsymbol{\Delta}_n^E$ is the extraclass distance for sample $\boldsymbol{x}_n$, w_n a sample weight defined as

$$w_n = \frac{\min\{\|\boldsymbol{\Delta}^E\|^\alpha, \|\boldsymbol{\Delta}^I\|^\alpha\}}{\|\boldsymbol{\Delta}^E\|^\alpha + \|\boldsymbol{\Delta}^I\|^\alpha} \tag{7}$$

and α is a control parameter between zero and infinity. The within-class scatter matrix is defined in the same way as FDA (eq.2).

Figure (1) shows the FDA and NDA solutions for two artificial datasets. For this example a single nearest neighbor was used in the computation of the between-class scatter matrix and uniform sample weights were considered. Particularly interesting is the case illustrated in fig. (1.b). Though both within-class scatter matrices are equal, the bimodality of one of the classes displaces the estimate of the class mean used in the computation of the parametric between-class scatter matrix. This is the main source of error for FDA.



(a) (b)

Fig. 1. First direction of nonparametric discriminant projection space on two artificial datasets. Dashed line: FDA direction. Solid line: NDA direction.

NDA and Nearest Neighbors. Making use of the introduced notation we can examine the relationship between NN and NDA. Given a training sample $\boldsymbol{x}$, the accuracy of the 1-NN rule can be directly computed by examining the ratio $\|\boldsymbol{\Delta}^E\|/\|\boldsymbol{\Delta}^I\|$. If this ratio is more than one, $\boldsymbol{x}$ will be correctly classified.

Given a $M \times D$ linear transform $\boldsymbol{W}$, the projected distances are defined as $\boldsymbol{\Delta}_{\boldsymbol{W}}^{E,I} = \boldsymbol{W}\boldsymbol{x} - \boldsymbol{W}\boldsymbol{x}^{E,I}$ Notice that this definition does not exactly agree with the extra and intraclass distances in projection space since, except for the orthonormal transformation case, we have no warranty on distance preservation. Equivalence of both definitions is asymptotically true on the number of samples. By the above remarks it is expected, that optimization of the following objective function should improve or, at least not downgrade NN performance,

$$\hat{\boldsymbol{W}} = \arg\max_{\boldsymbol{W}} \frac{E\{\|\boldsymbol{\Delta}_{\boldsymbol{W}}^{E}\|^2\}}{E\{\|\boldsymbol{\Delta}_{\boldsymbol{W}}^{I}\|^2\}} \tag{8}$$

Considering that [5], we have that

$$E\{\|\boldsymbol{\Delta}_{\boldsymbol{W}}^{E,I}\|^2\} = trace(\boldsymbol{W}^T \boldsymbol{S}_{b,w} \boldsymbol{W}) \tag{9}$$

where, in this case, $\boldsymbol{S}_b$ (the between-class scatter matrix) agrees with (6), but the within-class scatter matrix is now defined in a nonparametric fashion [6],

$$S_w = \frac{1}{N} \sum_{n=1}^{N} \Delta_n^I \Delta_n^{I\,T} \tag{10}$$

The same methodlogy that can be used to solve (1) can also be applied to the optimization of this objective function (8). This method has showed a good performance for standard data sets as well as for practical applications [6], but presents some problems when intraclass (or extraclass) diferences are not normally distributed around a direction.

2 A New Embedding Technique

In this section we propose the construction of a global discriminant embedding using discriminant projections that can be seen as the combination of multiple NDA projections. We are interested in a combination of one-dimensional projections that can yield a strong nearest neighbor classifier.

The main idea can be stated as follows: if we push the NDA approach to its limits, we can consider that every point $\boldsymbol{x}^j$ in the sample has associated *its most discriminant 1D-projection* $\boldsymbol{W}^j$, that is, the 1D projection that simultaneously minimizes the norm of the vectors that point to the nearest neighbor of its class and to the nearest neighbor of the other class[1]. Thus, given a learning problem,

[1] Observe that this concept can be easily defined to a m-class setting by considering the $m-1$ nearest neighbors in its class and the nearest neighbor in each one of the other classes.

we can get as much 1-D NDA projections as examples we have in the learning set. Our hypothesis is that a careful selection of a subset of these 1D-projections can define an embedding (where each new dimension is defined by a 1D projection) of the original data that outperforms the other discriminant methods when using the nearest neighbour classifier.

Our scheme takes benefit of a very known algorithm in machine learning, Adaboost ([8]), for selecting the best 1D NDA projections. The use of boosting in our scheme is specially justified, because our 1D projections perform always as weak classifiers (In fact, see figure 3, these classifiers have a similar performace to the nearest neighbor classifier in the original space), and we can exploit the sample weight actualization intrinsic in the boosting scheme to focus the selection of the next feature axis to the examples that are more difficult to classify.

Let $\boldsymbol{x}^k$ be a data point, $\boldsymbol{x}^i$ its nearest neighbor of the same class and $\boldsymbol{x}^e$ its nearest neighbor of the other class ($\boldsymbol{x}^k, \boldsymbol{x}^i, \boldsymbol{x}^e \in X$). We will define the vectors u and v which point to $\boldsymbol{x}^i$ and $\boldsymbol{x}^e$ from $\boldsymbol{x}^k$. We need to find a linear projection $f(x) : X \rightarrow R$ that minimizes the distance between the point $f(\boldsymbol{x}^k)$ to the points of its same class, and maximizes the distance to the points of the other class. In the case we are dealing with the projection matrix will be a simple vector that can be computed using simple vector operations.

2.1 AdaBoost

We have followed a boosting implementation similar to the one proposed by Viola et al. [7]. Given a training set of n points $\boldsymbol{x}^{1\ldots n}$ belonging to k different classes ($\frac{n}{k}$ points for each class), the algorithm performs as follows:

1. First we define a set of weights $\boldsymbol{W}^{1\ldots n}$ (each weight assigned to one vector). The weights are initialized to $\frac{1}{n}$. We also build the set of partial classifiers as 1D projections as defined above, so each sample $\boldsymbol{x}^i$ generates a projection to a 1D dimensional space.
2. Then a fixed number of boosting steps are generated. At each boosting step s:
   - The whole set of classifiers is tested using the training points $\boldsymbol{W}^{1\ldots n}$. We project each data point in the 1D space generated by each feature extraction and classify it according to it's nearest neighbor. For each different projection, we evaluate its classification error as:

$$Error_j = \sum_{i=1}^{n} W_{s,i} \cdot l_{i,j} \tag{11}$$

     where $l_{i,j}$ is set to 0 if the point x_i has been correctly classified by the classifier j and to 1 otherwise. Finally we select the classifier c with minimum $Error_{1..n}$
   - Using the classification results of the classifier c, the set of weights $\boldsymbol{W}$ is actualized as:

$$\boldsymbol{W}^{s+1,i} = \boldsymbol{W}^{s,i} \cdot \beta^{1-l_{i,c}} \tag{12}$$

where

$$\beta = \frac{Error_c}{1 - Error_c} \tag{13}$$

– The coefficient α_s corresponding to the classifier at the step s is computed as:

$$\alpha = log\frac{1}{\beta} \tag{14}$$

– Finally the weights are normalized, $\boldsymbol{W}^{s+1,i} = \frac{\boldsymbol{W}^{s,i}}{\sum_j^n \boldsymbol{W}^{s,j}}$.

3. The output of the algorithm is a projection matrix, where we place at each column i_s the 1-D projection corresponding to the best classifier at the step s of the Adaboost algorithm. In addition the $\alpha_{1,\ldots,s}$ coefficients can be used to rank the importance of the features extracted for each 1-D projection.

3 Application and Results

Cork inspection is the least automated task in the production cycle of the cork stopper. Due to the inspection difficulty of the natural cork material and the high production rates even the most experienced quality inspection operators frequently make mistakes. In addition, human inspection leads to a lack of objectivity and uniform rules applied by different people at different time. As a result, there is a urgent need to modernize the cork industry in this direction. In this paper, we consider a real industrial computer vision application of classification of natural (cork) products.

During its production, cork stoppers must be classified in five different classes that correspond to different quality groups (see fig. 2). When human operators perform this classification on-line, they rely on a set of visual characteristics that are far from being objective and that present a large variation among different operators. In order to develop an automatic system, a large set of carefully classified stoppers have been selected (more than one thousand examples per class). Next, we have got an image from every stopper that represents its surface, and this image has been segmented using a fixed threshold. Cork stopper classification will be based on a set of visual features that are related to the blobs resulting from this segmentation.

We have extracted from the image of each stopper a set of global as well local features [10]. Global features are: the total number of blobs, the total area of blobs, the mean of grey-level appearance of blobs, the average blob area, the average blob elongation, the average blob compactness, and the average blob roughness. Local stopper features refer to the first and second largest blobs of the cor stopper and particularly: area, length, width, perimeter, convex perimeter, compactness, roughness, elongation, average blob grey-level, and position with respect to the centre of the stopper. Following this strategy we defined a set of 43 features for every cork stopper.

Next, we have used this learning set for constructing a discriminant embedding as described in the last section. Figure (3) shows the result of the learning

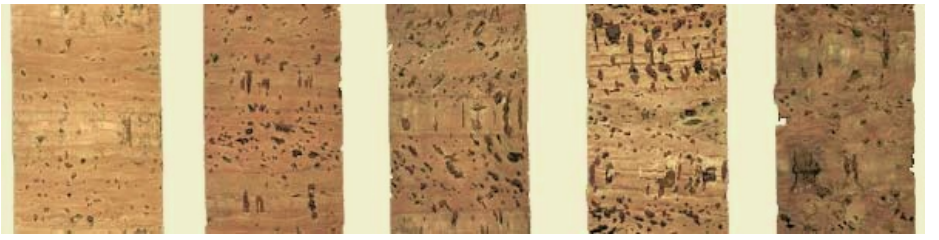


Fig. 2. Surfaces of cork stoppers of 5 quality groups ordered from best to worst quality (from left to right).

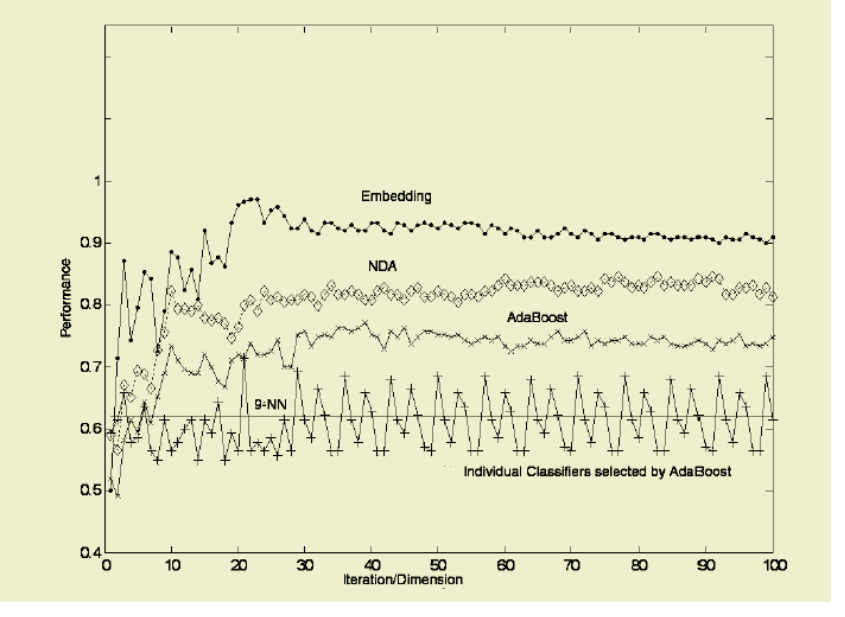


Fig. 3. Results: the horizontal line (solid line) represents the performance of a 9-nearest neighbor classifier in the original space, the + line represents the performace of every individual 1D classifier that is computed at every step of the algorithm, the × line corresponds to classifier that would be produced by the Adaboost combination, the ♢ line represents the NDA performance for different dimensionalities, and finally, the • line represents the performance of the nearest neighbor classifier in the embedding space.

method. Results have been computed with a 10-fold cross-validation, using a data set of 1000 samples per class. As can been seen, classifiying a stopper using the nearest neighbor in the embedded space shows the best performance when compared to the other methods: nearest neighbor in the original space, NDA of different dimensions, Adaboost classifier stopped at different iteration steps, and the set of 1D classifiers that are computed at every step of the Adaboost algorithm.The embedding approach converges, with respect to dimension, to a

90 per cent of correct classification, while all the other methods are all under or around 80 per cent.

4 Conclusions

We have presented a new method for learning a linear embedding for labeled data that is specially designed to be used with the nearest neighbor classifier. Every embedding dimension is defined by a linear projection that corresponds to the optimal projection of a given point. This projection is selected in a sound way by using the Adaboost algorithm. We have shown the performance of this method in a real industrial application: the quality classification of cork stoppers.

Acknowledgments

This work is supported by MCYT grant TIC2003-00654, Ministerio de Ciencia y Tecnologia, Spain.

References

1. R. Fisher: On subharmonic solutions of a Hamiltonian system. The use of multiple measurements in taxonomic problems, Ann. Eugenics 7 (1936) 179–188.
2. M. Aladjem: Linear discriminant analysis for two classes via removal of classification structure, IEEE Trans. Pattern Anal. Machine Intell. 19 (2) (1997) 187–192.
3. K. Fukunaga, J. Mantock: Nonparametric discriminant analysis, IEEE Trans. Pattern Anal. Machine Intell. 5 (6) (1983) 671–678. 11
4. P. Devijver, J. Kittler: Pattern Recognition: A Statistical Approach, Prentice Hall, London, UK, 1982.
5. K. Fukunaga: Introduction to Statistical Pattern Recognition, 2nd Edition, Academic Press, Boston, MA, 1990.
6. M. Bressan, J. Vitria: Nonparametric discriminant analysis and nearest neighbor classification, Pattern Recognition Letters 24 (15) (2003) 2743–2749.
7. P. Viola, M. Jones: Rapid object detection using a boosted cascade of simple features, in: IEEE Conference on CVPR, Kauai, Hawaii, 2001, pp. 511–518.
8. Y. Freund, R. E. Schapire: Experiments with a new boosting algorithm, in: International Conference on Machine Learning, 1996, pp. 148–156.
9. R. E. Schapire: A brief introduction to boosting, in: IJCAI, 1999, pp. 1401–1406.
10. P. Radeva, M. Bressan, A. Tobar, J. Vitrià: Bayesian Classification for Inspection of Industrial Products, in M.T. Escrig Monferrer, F. Toledo, E. Golobardes (Eds.), Topics in Artificial Intelligence, Sringer Verlag Series: Lecture Notes in Computer Science. Volume. 2504, 2002, pp. 399–407.