Choice of B-splines with free parameters in the flexible discriminant analysis context

Christelle Reynès\textsuperscript{a},*, Robert Sabatier\textsuperscript{a}, Nicolas Molinari\textsuperscript{b}

\textsuperscript{a}Laboratoire de Physique Moléculaire et Structurale, Faculté de Pharmacie, 15 av. Ch. Flahault, BP 14491, 34093 Montpellier Cedex 5, France
\textsuperscript{b}Laboratoire de Biostatistique, Institut Universitaire de Recherche Clinique, 641 av. G. Giraud, 34093 Montpellier, France

Received 28 February 2005; received in revised form 23 November 2005; accepted 24 November 2005
Available online 19 December 2005

Abstract

Flexible discriminant analysis (FDA) is a general methodology which aims at providing tools for multigroup non linear classification. It consists in a nonparametric version of discriminant analysis by replacing linear regression by any nonparametric regression method. A new option for FDA, consisting in a nonparametric regression method based on B-spline functions, will be introduced. The relevance of the transformation (hence the discrimination) depends on the parameters defining the spline functions: degree, number and location of the knots for each continuous variable. This method called FDA-FKBS (Free Knot B-Splines) allows to determine all these parameters without the necessity of many prior parameters. It is inspired by Reversible Jumps Monte Carlo Markov Chains but the objective function is different and the Bayesian aspect is put aside.

© 2005 Elsevier B.V. All rights reserved.

Keywords: Discrimination; Flexible discriminant analysis; Reversible jump; Monte Carlo Markov chain; B-splines

1. Introduction

Additive splines have already been used to perform discriminant analysis (Cornillon et al., 2003; Durand, 1992). But applying it on very complex data is not very easy for the operator has to fix the spline parameters: degree, number and location of knots. Otherwise, several methods have been constructed to find good spline parameters and they have been used in curve-fitting (DiMatteo et al., 2001; Durand and Sabatier, 1997; Jupp, 1978; Lindstrom, 1999). One can notice that they only focus on the determination of knots and not of the degree. Indeed, the spline nature makes them able to fit numerous and very complex forms. In this article, we introduce an algorithm, inspired by Reversible Jumps Monte Carlo Markov Chains (RJMCMC) (Green, 1995), to use free-knot B-splines as an addition to the existing tool kit for the flexible discriminant analysis (FDA) (Hastie et al., 1994). Indeed, FDA is a multigroup nonlinear discrimination method, which is based on nonparametric regression followed by linear discriminant analysis (LDA). LDA is a very common and useful method which calls for simple and rather intuitive concepts. It consists in looking for linear functions which split as well as make possible the projections of the different observations into the different groups and which
gather the projections of the individuals belonging to the same group. But, this way of proceeding is unable to find out complex structures because it supposes linear limits between groups. We introduced the spline functions in order to take into account the potential non-linearity before LDA application. In fact, the spline transformation consists in an increase of the data dimension and then, we use LDA to reduce it.

Spline functions are used to find a relation between the variables describing the groups and the classes. Then, we apply LDA on the new array where rows are the individuals and columns the images of the original variables with the spline functions. In this context, the determination of the spline parameters is very difficult because the transformation of each variable has to take into account the different groups. That is why we had recourse to an existing method that allows dimension changes. Indeed, our method is distantly inspired from Reversible Jump Markov Chain Monte Carlo (RJMCMC) (Green, 1995). RJMCMC is used for model determination where the dimensionality of the parameter vectors is not fixed. Hence, it is particularly well adapted to free-knot splines (DiMatteo et al., 2001; Hansen and Kooperberg, 2002; Lindstrom, 1999). We kept the general principle of RJMCMC but we put the Bayesian aspect aside. Let us notice that Bayesian methods have already been used to determine parameters in penalized B-splines in regression context (Brezger and Lang, 2006). Thus, we constructed a method we will call FDA-FKBS (FDA-Free Knot B-Splines).

We will begin our statement with a little reminder about FDA, spline functions and especially free-knot B-splines. Then, our methodology will be presented and applied on three data sets. The quality of our algorithm will be tested by assigning scores to the classes such that the transformed class labels are optimally predicted by regression.

2. Flexible discriminant analysis

FDA (Hastie et al., 1994, 2001) is a discriminant method that performs LDA on derived responses. Those responses are obtained by assigning scores to the classes such that the transformed class labels are optimally predicted by regression on X. We suppose \( \mathcal{G} \) is the class of the \( t \)th sample and \( \mathcal{G}(>1) \) groups

This analysis will be based on the observation of \( p(>1) \) quantitative variables \( X^1, X^2, \ldots, X^p \) and one qualitative variable \( Y \) called response. The response gives the class to which each sample belongs. We consider \( G(>1) \) groups such that the response is valued on \( \mathcal{G} = \{1, \ldots, G\} \). \( (n_1, n_2, \ldots, n_G) \) are the numbers of observations in each group, we have got \( n = \sum_{g=1}^{G} n_g \) observations. We will denote by \( X \) the matrix \((n \times p)\) whose columns are the vectors \((X^1, X^2, \ldots, X^p)\). Let \( A \) be the \((n \times G)\) indicator dummy matrix of the response. We denote by:

- \( I_k \) the identity matrix having \( k \) columns and \( k \) rows,
- \( X' \) the transposed matrix of \( X \),
- \( P_X \) the orthogonal projector onto the space spanned by \( X \) columns.

\[
\begin{align*}
\min_{\beta, \theta} \|A\theta - X\beta\|^2 \\
\text{with conditions on } \theta: \text{mean zero and unit variance. Here we assume that } p \geq G. \text{As in LDA, we can generalize it and define } L \leq G - 1 \text{ vectors } \Theta = (\theta_1, \theta_2, \ldots, \theta_L). \text{The set of scores are assumed to be mutually orthogonal and normalized with respect to an appropriate inner product to prevent trivial zero solutions. We have to construct } L \text{ corresponding vectors } \beta_i \text{ for } i \in \{1, \ldots, L\}. \text{Then, the discrimination criterion is the average squared residual defined by}
\end{align*}
\]

\[
\text{ASR} = \frac{1}{n} \sum_{i=1}^{L} \sum_{i=1}^{n} \left( \theta_i (g_i) - x_i' \beta_i \right)^2,
\]

where \( g_i \) is the class of the \( i \)th sample and \( x_i \) is the vector of \( p \) predictors for the \( i \)th sample.

But FDA is far more than that. Indeed, \( x_i' \beta_i \) indicates a linear regression and it can be replaced by non-parametric regression \( \eta_i(x_i) \). This will be useful for overcoming the drawbacks of linear separations in LDA. Then, the criterion
Thus, FDA can be considered as the application of LDA on the matrix obtained with the nonparametric regression and on the transformed class matrix.

The computational steps are the following ones:

1. **Initialize.** Choose an initial $G \times L$ ($L \leq G - 1$) score matrix $\Theta_0$ satisfying the constraint $\Theta_0' D_\rho \Theta_0 = I_L$ where $D_\rho = A' A / n$; let $\Theta_0^* = A \Theta_0$ be the transformed responses.

2. **Multivariate nonparametric regression.** Fit a multiresponse adaptive nonparametric regression of $\Theta_0^*$ on $X$. Let $S(\hat{\lambda})$ be the linear operator that fits the final chosen model with $\hat{\lambda}$ the vector of optimized tuning parameters for this model and let $\eta(x)$ be the vector of fitted regression functions.

3. **Optimal scores.** Obtain the eigenvector matrix $\Phi$ of $\Theta_0^* S(\hat{\lambda}) \Theta_0^*$, and hence the optimal scores $\Theta = \Theta_0 \Phi$.

4. **Update** the final model from step 2 using the optimal scores: $\eta(x) \leftarrow \Phi' \eta(x)$.

In fact, it is possible to express LDA in the same computational way:

1. Form $A$, the $n \times G$ indicator matrix.
2. Set $\hat{A} = P_X A$ and denote by $B$ the coefficient matrix: $\hat{A} = X B$.
3. Obtain the eigenvector matrix $\Theta$ of $\hat{A}' P_X A$ with normalization $\Theta' D_\rho \Theta = I_L$.
4. Update the coefficient matrix in step 2 to reflect the optimal scores: $B \leftarrow B \Theta$. The final optimally scaled regression fit is the vector function $\eta(x) = B' x$.

Thus, FDA can be considered as the application of LDA on the matrix obtained with the nonparametric regression and on the transformed class matrix.

### 3. B-spline functions

We will use the B-splines because they are well adapted to numerical calculation, especially in the case of multiple knots, that is to say several knots in the same place (de Boor, 1978). To define the B-spline function, we have to make a subdivision of the interval $[a^i, b^i]$ on which the variable $X^i$ ($i = 1, \ldots, p$), we want to transform, is valued. Let

$$
\begin{align*}
(r_0^i) &= a^i < r_1^i < r_2^i < \cdots < r_j^i < b^i \quad (= r_{j+1}^i),
\end{align*}
$$

be this subdivision. These $J^i$ points are denoted by breaks. The spline function $s^i(X^i)$ used to transform $X^i$ is a polynomial of degree $d^i$ on any of the $J^i + 1$ intervals $[r^i_{j-1}, r^i_j]$. Each break corresponds to one or more knots leading to the knot sequence $\zeta_1^i \leq \zeta_2^i \leq \cdots \leq \zeta_{K^i}^i$, the number of knots for one break is called its multiplicity. The multiplicity defines the continuity conditions at the break. Indeed, for one simple knot there are $d^i - 1$ continuous derivatives on the considered interval, for a double knot there are $d^i - 2$ continuous derivatives. So that the function and its $d^i$ derivatives are discontinuous at the boundaries, $d^i + 1$ knots are systematically placed at the lower and upper bounds of the interval, here $a^i$ and $b^i$. Thus, as we will see later, the knot multiplicity is very important.

For each fixed sequence of knots $\zeta_s^i = (\zeta_1^i, \zeta_2^i, \ldots, \zeta_{K^i}^i)$, the set of such splines is a linear space of functions with $K^i + d^i + 1$ free parameters (de Boor, 1978). Hence, to specify a B-spline function one has to specify its order $(d^i + 1)$, the number of knots and their location.

A useful basis $(B^i_{l})_{l=1,\ldots,K^i+d^i+1}$ for this linear space is given by Schoenberg’s B-splines (Curry and Schoenberg, 1966). Then, we can write a spline as

$$
\begin{align*}
s^i(X^i, \beta^i, \zeta^i, d^i) &= \sum_{l=1}^{K^i+d^i+1} \beta^i_l B^i_l \left( X^i, \zeta^i, d^i \right),
\end{align*}
$$

(4)
where \( \beta^i = (\beta^i_1, \ldots, \beta^i_{K^i+d^i+1})' \), the vector of coefficients and the vector \( \xi^i \) of knots are considered as tuning parameters.

In our regression context, we can estimate \( \beta^i \) with the usual least square method:

\[
\hat{\beta}^i = \left( \hat{\beta}^i_1, \ldots, \hat{\beta}^i_{K^i+d^i+1} \right)'
= \left( B^i \left( X^i, \xi^i, d^i \right) B^i \left( X^i, \xi^i, d^i \right) \right)^{-1} B^i \left( X^i, \xi^i, d^i \right)' A,
\]

(5)

where the matrix \( B^i \left( X^i, \xi^i, d^i \right) \) is the encoding matrix of \( X^i \) in B-spline basis which is defined by: \( B^i \left( X^i, \xi^i, d^i \right) = \left[ B^i_1 \left( X^i, \xi^i, d^i \right), \ldots, B^i_{K^i+d^i+1} \left( X^i, \xi^i, d^i \right) \right] \). It is a \( \left[ n \times (K^i + d^i + 1) \right] \) matrix. If \( \left( B^i \left( X^i, \xi^i, d^i \right) \right)^{-1} \) does not exist, we will use the Moore–Penrose inverse denoted by \( \left( B^i \left( X^i, \xi^i, d^i \right) \right)^{+} \).

### 3.1. Free knot splines

Here, we consider the \( i \)th variable, in the case of known number of knots and degree. The location of knots is the only free parameter. We aim at finding the response value contained in \( A \), consequently, we can use the least square approach and try to find

\[
\min_{\beta^i \in \mathbb{R}^{K^i+d^i+1}} \min_{\xi^i \in [0, b)^{K^i}} \left\| A - B^i \left( X^i, \xi^i, d^i \right) \beta^i \right\|^2,
\]

(6)

where \( \left\| A - B^i \left( X^i, \xi^i, d^i \right) \beta^i \right\|^2 = \text{tr} \left( A - B^i \left( X^i, \xi^i, d^i \right) \beta^i \right) \left( A - B^i \left( X^i, \xi^i, d^i \right) \beta^i \right)' \).

We already know that we can estimate \( \beta^i \) for fixed \( \xi^i \) by

\[
\hat{\beta}^i \left( \xi^i \right) = \left( B^i \left( X^i, \xi^i, d^i \right)' B^i \left( X^i, \xi^i, d^i \right) \right)^{-1} B^i \left( X^i, \xi^i, d^i \right)' A,
\]

then

\[
F \left( \xi^i, d^i \right) = \left\| A - B^i \left( X^i, \xi^i, d^i \right) \beta^i \left( \xi^i, d^i \right) \right\|^2 = \left\| A - B^i \left( X^i, \xi^i, d^i \right) \left( B^i \left( X^i, \xi^i, d^i \right)' B^i \left( X^i, \xi^i, d^i \right) \right)^{-1} B^i \left( X^i, \xi^i, d^i \right)' A \right\|^2
= \left\| A - P^i_{B^i} \left( X^i, \xi^i, d^i \right)' A \right\|^2,
\]

where \( P^i_{B^i} \left( X^i, \xi^i, d^i \right) \) is the orthogonal projector onto the space spanned by \( B^i \left( X^i, \xi^i, d^i \right) \) columns.

Then, minimizing \( F_1 \left( \xi^i, \beta^i, d^i \right) \) is equivalent to minimizing \( F \left( \xi^i, d^i \right) \) (Golub and Pereyra, 1973). But achieving this is a hard task. The most important difficulties are summarized in the lethargy property (Jupp, 1975) which has two main consequences in our context. Firstly, this kind of problem is linked with numerous local minima. That is why any algorithm will probably not converge to the global optimum if it is poorly initialized. However, we cannot have any idea of how to initialize the algorithm. Thus, it is a very big difficulty. Secondly, the knots will be likely to coalesce, that is to be in the same place. This is something we generally try to avoid. Now, let us see the method we developed in order to reduce these drawbacks.
4. FDA-FKBS

4.1. General principle

We saw in the previous paragraph that if we find $\zeta$ and $d$ minimizing $F(\zeta, d)$, we will minimize the criterion of the linear regression of $A$ on $B(\zeta, d)$ (Eq. (6)). Then, we can write a simplified expression of $F(\zeta, d)$:

$$ F(\zeta, d) = \|A - B(\zeta, d)(B(\zeta, d)'B(\zeta, d))^{-1}B'(\zeta, d)A\|^2. $$(7)

Now, let us connect this criterion with FDA. With the spline transformation step, we perform a nonparametric regression of $A$ on $X$. But, even if the parameters are well determined, we know that the regression of an indicator matrix is not very efficient for discrimination, especially when there are more than two groups (Hastie et al., 2001). That is why we apply LDA on the transformed variables and then, we can describe our method as another option for the FDA method.

1. We choose an initial score matrix: $\Theta_0 = I_L$ then $\Theta_0^* = A$.
2. We perform our FDA-FKBS method (see next paragraph) to find $\hat{\zeta}$ and $\hat{d}$, that is we fit a multiresponse adaptive non-parametric regression of $\Theta_0^* = A$ on $X$. The linear operator that fits the final chosen model is $S(\hat{\zeta}, \hat{d}) = B(\hat{\zeta}, \hat{d})(B(\hat{\zeta}, \hat{d})'B(\hat{\zeta}, \hat{d}))^{-1}B'(\hat{\zeta}, \hat{d}) = P_B(\hat{\zeta}, \hat{d})$. Let $\eta(x)$ be the vector of fitted regression functions.
3. We obtain the eigenvector $L \times L$ matrix $\Phi$ of $\Theta_0^* S(\hat{\zeta}, \hat{d}) \Theta_0^*$, hence the optimal scores $\Theta = \Theta_0 \Phi$.
4. Update the final model from step 2 using the optimal scores: $\eta(x) \leftarrow \Phi' \eta(x)$.

$F(\zeta, d)$ will be our objective function. It is obviously different from the one used in RJMCMC (which is based on Bayesian aspects) but it was chosen to fit the spline context and was confirmed to be in keeping with the FDA methodology.

4.2. FDA-FKBS method of parameter selection

The determination of the spline parameters is the most difficult issue of our FDA method. Some methods allow one to find the knot locations given their number and the degree. We attempted to find an algorithm allowing deducing all the parameters. The following one is inspired from RJMCMC (Green, 1995) but is quite far from it. The objective of such methods is to be able to choose between different models when the dimensionality is variable. At one instance, we might want to find the number of hidden states in a hidden Markov model (Boys and Henderson, 2001). In our context, the number and location of knots and the degree are tuning parameters. RJMCMC has already been used in the same context (DiMatteo et al., 2001; Lindstrom, 2002), but, we add freedom on the degree. Moreover, our adaptation will be very different since we do not consider a Bayesian objective but an objective adapted from the function $F(\zeta)$, we saw in Section 4.1.

However, we will follow the general unfolding of RJMCMC. The first step of such a method is the definition of the different moves authorized to go from one model to another one. In our context, five possible moves can be defined: addition, deletion and relocation of a knot, increase and decrease of the degree. For each movement, we have to put priors (Green, 1995). Then, at each step, we randomly choose the variable on which we will perform the move. We compute a probability for each move and according to these probabilities, we choose the move we will propose. That is what we explain now:

- **Knot deletion**: If we have $k$ knots for the chosen variable, the probability to remove one of the knots is

$$ p_{del}(k) = c \min \left(1, \frac{q_1(k-1)}{q_1(k)}\right), $$

where $q_1(k)$ is the prior probability to have $k$ knots, $c$ is a constant which keeps the sum of all the probabilities less than 1 (see the probability of relocating a knot). As DiMatteo et al. (2001) proposed, we chose Poisson(3) prior...
for \( q_1 \). With this solution, we increase the probability for small number of knots and hence, we support the model parsimony. If this move is chosen, we randomly pull out one of the existing knots and we remove it.

- **Knot addition:** In the same way, if we have \( k \) knots for the chosen variable, the probability to add one knot is

\[
p_{\text{add}}(k) = c \min \left( 1, \frac{q_1(k + 1)}{q_1(k)} \right).
\]

If this move is chosen, we randomly appoint one of the existing knots. This knot will be the pivot of the distribution law we will use to locate the new knot.

- **Degree increase:** If the degree is \( d \) for the chosen variable, the probability to increase it is

\[
p_{\text{inc}}(d) = 0.3 \min \left( 1, \frac{q_2(d + 1)}{q_2(d)} \right),
\]

where \( q_2(d) \) is the prior probability to have degree \( d \), here, it is a Poisson(2) prior. We multiply by 0.3 to make the probability lower, in order to support the moves concerning knots. Indeed, the possible field for knots is far larger, as it concerns the number and the location. The probability of increasing the degree will approximately be three times lower than the probability of adding a knot. Let us notice that if this move is chosen, the degree will become \( d + 1 \).

- **Degree decrease:** Alike, if the degree is \( d \) for the chosen variable, the probability to decrease the degree is

\[
p_{\text{dec}}(d) = 0.3 \min \left( 1, \frac{q_2(d - 1)}{q_2(d)} \right).
\]

- **Knot relocation:** If we have \( k \) knots and degree \( d \) for the chosen variable, the probability to relocate a knot is

\[
p_{\text{rel}} = 1 - p_{\text{add}} - p_{\text{del}} - p_{\text{inc}} - p_{\text{dec}}.
\]

If this move is chosen, we randomly appoint the knot we will relocate, and we define its new location by a probability law centred on its previous place.

FDA-FKBS was proved to be robust when the parameters are modified. Once the move is chosen and applied, we do not accept all the proposed moves. Indeed, an acceptance probability, \( p_{\text{acc}} \), is defined for each move:

\[
p_{\text{acc}} = \min \left( 1, \left( \frac{\text{obj}_{\text{prev}}}{\text{obj}_{\text{cand}}} \right)^{p_1} \times \left( 1 - \frac{4(k_{\text{prev}} + d_{\text{prev}})}{n} \right)^{p_2} \right).
\]

This probability takes into account the objective function (defined in Eq. (7)) obtained with the last accepted model, \( \text{obj}_{\text{prev}} \), and with the proposition (the candidate) we are studying, \( \text{obj}_{\text{cand}} \). But the model has to be parsimonious (to avoid over-fitting), so we propose a penalization based on the one presented in Pittman (2002). This criterion is founded on GCV which was originally stated by Craven and Wahba (1979). Pittman (2002) proposed to penalize the objective with different parameters: \( k_{\text{prev}} \) and \( d_{\text{prev}} \), the number of knots and the degree of the last accepted model, \( k_{\text{cand}} \) and \( d_{\text{cand}} \), the number of knots and the degree of the proposition we are studying and the population size, \( n \).

We also use \( p_1 \) and \( p_2 \) in order to create a balance between the model parsimony and precision. Those constants are determined by two functions depending on the objective value for \( p_1 \) and on \( n \) for \( p_2 \). We observed many steps of our algorithm and we found that a linear function between \( p_1 \) and the objective gave good results, concerning acceptance probabilities. We noticed the same phenomenon between \( p_2 \) and \( n \). Hence, we did a regression between \( p_1 \) and the objective values that seems to give good acceptance probabilities and we did the same for \( p_2 \). We tested those functions for several data sets and it appears to be a good solution.

All those parameters allow us to perform a succession of models as we see in the following algorithm. This algorithm has been implemented in R© (R Development Core Team, 2004) and Matlab 7 (MATLAB, 2004).
FDA-FKBS general unfolding
1. Generate initial model with maximum number of knots and degree for each variable
2. while stopping criteria not met do
3. random choice of the variable whose parameters are likely to be modified
4. calculation of each move probability
5. choice and application of the move
6. calculation of the acceptation probability
7. acceptation or refusal
8. if acceptation do potential recording of the new parameters
9. end while

Concerning the initial model, we tested maximum, minimum and random parameters. But the algorithm was not sensitive to this and we chose the maximum number of knots and degree in order to see the way to parsimony. In our case, the stopping criteria is the maximum number of iterations. Indeed, we run 100,000 propositions saving every 10th update after the first 10,000 (burning period) (Lindstrom, 2002).

At the end of the algorithm, we have to make a posterior determination of the parameters. We keep the most often obtained combination which corresponds to the mode of the posterior distribution. As the parameters are completely free, we have to perform numerous iterations. In fact, we are able to get a good result in 15 to 40 min. It depends on the number of variables and on the data complexity.

5. Application to simulated data

5.1. The disc data

5.1.1. Data construction

We aim at making a discrimination, thus we need data split into groups and described by quantitative variables to make the spline transformation. The Disc 1 data set is inspired by Zhu and Hastie (2003). It consists of three groups described by 10 variables. The original data set had 20 variables but 10 variables are enough to show what we want. Indeed, the first two variables determine the space location of the different groups shown in Fig. 1, whereas the eight other variables are noise. The first two variables are constructed as follows:

- We generate five independent variables \( \theta_1, \ldots, \theta_5 \) uniformly distributed on \([0; 2\pi]\), with 100 observations for each one.
- We generate error terms \( \varepsilon_1, \ldots, \varepsilon_{10} \) which are i.i.d. drawn from normal distribution \( N(0, 1) \).
- First group:
  - abscissas: \( gp1X = \sin \theta_1 + \varepsilon_1 \),
  - ordinates: \( gp1Y = \cos \theta_1 + \varepsilon_2 \).
- Second group:
  - first subgroup abscissas: \( gp21X = \sin \theta_2 + \varepsilon_3 + 10 \),
  - first subgroup ordinates: \( gp21Y = \cos \theta_2 + \varepsilon_4 \),
  - second subgroup abscissas: \( gp22X = \sin \theta_3 + \varepsilon_5 - 10 \),
  - second subgroup ordinates: \( gp22Y = \cos \theta_3 + \varepsilon_6 \).
- Third group:
  - first subgroup abscissas: \( gp31X = \sin \theta_4 + \varepsilon_7 \),
  - first subgroup ordinates: \( gp31Y = \cos \theta_4 + \varepsilon_8 + 10 \),
  - second subgroup abscissas: \( gp32X = \sin \theta_5 + \varepsilon_9 \),
  - second subgroup ordinates: \( gp32Y = \cos \theta_5 + \varepsilon_{10} - 10 \).

The plane obtained with these variables is the one shown in Fig. 1.

Then, the 8 other variables are i.i.d. drawn from normal distribution \( N(0, 1) \) (they also have 100 observations).
5.1.2. Discrimination

In order to situate the efficiency of our method, we applied eight other different discriminant methods: LDA, quadratic discriminant analysis (QDA), k-nearest neighbours (k-nn), classification tree, polyclass (Kooperberg et al., 1997), polymars (Kooperberg et al., 1997) and, obviously, flexible discriminant analysis (FDA) with two regression methods used in optimal scaling: Multivariate adaptive regression splines (MARS) (Friedman, 1991) and BRUTO (Hastie et al., 1994). It is interesting to notice that MARS has recently and successfully been used for discrimination in a credit scoring context (Lee et al., 2006). There is one function for each method in R:\(^\text{\texttt{qda()}}\), \texttt{tree()}, \texttt{knn()}, \texttt{polyclass()}, \texttt{polymars()}, \texttt{fda(method = mars)}, \texttt{fda(method = bruto)}, respectively. We can notice that we optimized the number of neighbours for k-nn, the maximum number of model terms for FDA-MARS and the cost for FDA-BRUTO. The optimization was achieved through cross-validation. The results for these methods and for FDA-FKBS are shown in Table 1. As we explained in the introduction, the prediction quality has been measured with five-fold cross-validation.

LDA results are very poor. We globally obtain 38% of good classifications in a prediction context. Furthermore, polyclass and polymars perform rather bad. In fact, they both use degree one splines. When we studied the R outputs, we noticed that polyclass did not put any knot on any variable. Thus, we removed all the noise variables and knots
Fig. 2. Representation of the Disc 2 data: the first group (+), the second (△) and the third one (●).

Table 2
Global rate of good classification for Disc 2 data

<table>
<thead>
<tr>
<th>Method</th>
<th>QDA</th>
<th>k-nn</th>
<th>Tree</th>
<th>FDA-MARS</th>
<th>FDA-BRUTO</th>
<th>FDA-FKBS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good prediction rate</td>
<td>0.84</td>
<td>0.88</td>
<td>0.87</td>
<td>0.89</td>
<td>0.90</td>
<td>0.90</td>
</tr>
</tbody>
</table>

appeared, giving perfect results. Then, we successively add noise variables. With one noise variable, problems arise: depending on the training set, we can obtain various results, either only knots on the meaningful variables (100% of good predictions), either one knot on one meaningful variable and one on the noise variable (80% of good predictions) or no knot at all (40% of good predictions). With two noise variables, most of the time, polymars does not find any knot (40% of good predictions). With polymars, the phenomenon is nearly the same.

However, the other methods are all very efficient with no error except for QDA in the first group. It can be explained by the very neat separation between groups. That is why we perform the analysis again with a bigger noise (Disc 2). Indeed, we generate error terms $\varepsilon_1, \ldots, \varepsilon_{10}$ which are i.i.d. drawn from normal distribution $N(0, 3)$. Then, there are many overlaps between groups as shown in Fig. 2. The global results of the discrimination are put in Table 2.

We see that, with these data, the methods are not equivalent. Indeed, the best ones are FDA-BRUTO, FDA-MARS and FDA-FKBS. The other methods perform quite well, but QDA is less robust when the data are less precise. Moreover, we can notice that with FDA-FKBS, the algorithm never locates any knot on the last eight variables which are only noise. It is very interesting and shows the efficiency of our algorithm in finding meaningful knots which can be interpreted. Furthermore, FDA-FKBS converges quite quickly as shown in Fig. 3. We see the burning-period before the line.

5.2. The circle data

5.2.1. Data construction

The circle data set is composed of three groups. We simulate three circles whose radii and centres are different. They are constructed as follows:

- We generate three independent variables with 100 observations for each one, $\theta_1, \theta_2$ and $\theta_3$ uniformly distributed on $[0; 2\pi]$,
Fig. 3. Evolution of the objective function along iterations.

Table 3
Rate of good classifications with the different methods for circle data

<table>
<thead>
<tr>
<th>Method</th>
<th>Group 1</th>
<th>Group 2</th>
<th>Group 3</th>
<th>Global</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA</td>
<td>0.14</td>
<td>0.39</td>
<td>0.66</td>
<td>0.33</td>
</tr>
<tr>
<td>QDA</td>
<td>0.61</td>
<td>0.67</td>
<td>0.68</td>
<td>0.65</td>
</tr>
<tr>
<td>k-nn</td>
<td>0.75</td>
<td>0.87</td>
<td>0.82</td>
<td>0.81</td>
</tr>
<tr>
<td>Tree</td>
<td>0.74</td>
<td>0.64</td>
<td>0.68</td>
<td>0.69</td>
</tr>
<tr>
<td>Polymars</td>
<td>0.72</td>
<td>0.68</td>
<td>0.64</td>
<td>0.68</td>
</tr>
<tr>
<td>Polymars</td>
<td>0.74</td>
<td>0.65</td>
<td>0.72</td>
<td>0.70</td>
</tr>
<tr>
<td>FDA-MARS</td>
<td>0.76</td>
<td>0.83</td>
<td>0.80</td>
<td>0.79</td>
</tr>
<tr>
<td>FDA-BRUTO</td>
<td>0.73</td>
<td>0.69</td>
<td>0.84</td>
<td>0.75</td>
</tr>
<tr>
<td>FDA-FKBS</td>
<td>0.79</td>
<td>0.78</td>
<td>0.85</td>
<td>0.81</td>
</tr>
</tbody>
</table>

- We generate error terms $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_6$ which are i.i.d. drawn from normal distribution $N(0, 0.1)$,
- First circle abscissas: $\text{disc1}X = \sin \theta_1 + \varepsilon_1$,
- First circle ordinates: $\text{disc1}Y = \cos \theta_1 + \varepsilon_2$,
- Second circle abscissas: $\text{disc2}X = \frac{1}{2} \sin \theta_2 + \varepsilon_3$,
- Second circle ordinates: $\text{disc2}Y = \frac{1}{2} \cos \theta_2 + \varepsilon_4$,
- Third circle abscissas: $\text{disc3}X = \frac{1}{2} \sin \theta_3 + \min(\text{disc1}X) + \frac{3}{4}(\max(\text{disc1}X) - \min(\text{disc1}X)) + \varepsilon_5$,
- Third circle ordinates: $\text{disc3}Y = \frac{1}{2} \cos \theta_3 + \varepsilon_6$.

Fig. 4 shows this data set.

5.2.2. Discrimination
We notice in Table 3 that FDA-FKBS and $k$-nearest neighbours are the best methods with regard to the prediction quality. For these methods, we predict the good class for more than four out of five observations. Moreover,
FDA-FKBS results are the most stable between groups and it provides a precise model which is not the case with $k$-nearest neighbours as it is a non-parametric method. We can also notice that we dramatically improve the results obtained with LDA. Indeed, with LDA, the results are those we should obtain by randomly affecting the class labels.

If we consider Bruto and MARS options for FDA, the results are quite satisfying and not very far from FDA-FKBS. However, these methods both provide numerous knots. Indeed, we obtained 104 knots per variable with Bruto and 71 with MARS. This is not very surprising because Bruto, for example, is based on smoothing splines which begin with one knot per input value, so the principles are completely different. However, the kind of model we provide is much more interpretable as can be seen by looking at the knot positions. With FDA-FKBS for these data, we frequently obtained two mingled knots (around $-0.13$). We said that we try to avoid this coalescence but let us study what are the consequences of such knots by considering the reconstruction of the second group for one of the five sub-samples (the results are nearly the same for each of the five sub-samples).

We see in Fig. 5 that the discontinuity due to the double knot is very useful for reconstituting the second group. There is a kind of jump on the double knot which keeps the individuals of group 2 above the limit of 0.5 and the individuals of group 3 under that limit. This example proves that it is not relevant to systematically avoid knot coalescence. Moreover, if we remove one of the knots (obtaining a single knot instead of the double one), we can notice that the results deteriorate (72% of good classifications). Hence, the double knot causes an improvement in the prediction quality with FDA-FKBS.

6. Application: Fish data

6.1. The data

This data set gathers morphological data concerning 120 fishes from species *Thymallus Thymallus*. They were firstly described in Persat (1978) and freely available in the software ADE 4 (Thioulouse et al., 1997). The individuals were caught in five different places which constitute the groups. The features are 13 morphological measures.
6.2. Discrimination

As previously, the same methods of discrimination were applied to the data and the results can be found in Table 4. These results were obtained by applying five-fold cross-validation. It is very different from the previous examples. Indeed, FDA-FKBS still gets satisfying results. Moreover, the model we obtained is quite simple as the spline degree is always one and the algorithm sets no knot on most of the variables; there are, on average, five knots spread over the 13 variables. Concerning the other options for FDA, they perform rather bad. They seem to be able to discriminate the fifth group from the other ones, affecting nearly all the other samples to the first group (which is the biggest one). The fifth group is indeed very different and all methods, except QDA, discriminate it very well. However, a few methods are able to discriminate quite well all the groups: LDA and FDA-FKBS are the best ones, FDA-FKBS gets more homogeneous results between groups. The following method in performance, polymars, gets less interesting results. It can be noticed that the results obtained by \( k \)-nn are not very satisfying, whereas it is a quite robust method which gets very good results in the other experiments.

In this example, we can study the convergence of the final population by looking at the knot locations for one variable in the last population. We chose the 10th variable on which FDA-FKBS systematically places one or two knots.
Fig. 6 shows that even if the range of this variable is \([-23.74, 26.26]\), the locations of the knots are not scattered but well gathered around two values, approximately \(-2.9\) and 0. This graph illustrates the convergence of the algorithm.

7. Conclusion

This article shows the efficiency of the combination between B-spline transformation of the variables and LDA. Indeed, our method allows completely free spline parameters. Hence, it is not necessary to know any precise thing about the data before performing FDA-FKBS. The cost of it is time, especially when we have numerous variables: we need at least 15 min to be able to get a good result. This drawback could perhaps be overcome by using another programming language.

When we compare with other discriminant methods, we notice that FDA-FKBS is always in the best ones. We can selectively find one of the other methods that is very efficient for one data set but our method seems to be very versatile. We can add that with this method we always found a limited number of knots: most of the time, we have between zero and five knots. As regards degree, it rarely exceeds 2. This observation shows the ability of our algorithm to maintain the parsimony. Moreover, we saw that the knots seem to be meaningful. Indeed, FDA-FKBS did not locate any knot on noise variables and it was able to find a double knot when it was useful to make the discrimination better. We also saw that coalescence is not always something to avoid but if the context excludes it, we can easily introduce, in our algorithm, something to forbid or to penalize it.

Let us notice that we initialized FDA-FKBS with the maximum parameters, but it is obviously possible to begin with no knots and degree 1 or random parameters. But, whatever the initial parameters, the algorithm converges to very near solutions. Furthermore, in this way, we can see again the way to parsimony.

In this work, we focused on LDA as a method for dimension reduction but we could attempt to use other methods. At last, as the location of knots seems to be relevant it could lead us to apply our method in regression contexts. All the R \(^\text{\textcopyright}\) (R Development Core Team, 2004) and MATLAB (MATLAB, 2004) codes used for the analysis can be obtained by writing to the corresponding author.
Acknowledgements

We are grateful to Trevor Hastie and Robert Tibshirani for their assistance with the use of FDA-Mars and FDA-Bruto in R. We also thank the editors and anonymous referees for helpful comments and suggestions.

References


