dawai: An R Package for Discriminant Analysis With Additional Information

David Conde
Universidad de Valladolid

Miguel A. Fernández
Universidad de Valladolid

Bonifacio Salvador
Universidad de Valladolid

Cristina Rueda
Universidad de Valladolid

Abstract

The incorporation of additional information to discriminant rules is receiving increasing attention as the rules including this information perform better than the usual rules. In this paper we introduce an R package called **dawai**, which provides the functions that allow to define the rules that take into account this additional information expressed in terms of restrictions on the means, to classify the samples and to evaluate the accuracy of the results. Moreover, in this paper we extend the results and definitions given in previous papers (Fernández *et al.* (2006), Conde *et al.* (2012), Conde *et al.* (2013)) to the case of unequal covariances among the populations, and consequently define the corresponding restricted quadratic discriminant rules. We also define estimators of the accuracy of the rules for the general more than two populations case. The wide range of applications of these procedures is illustrated with two data sets from two different fields such as biology and pattern recognition.

Keywords: classification rules, order-restricted inference, true error rate, R package **dawai**, R.

1. Introduction

The incorporation of additional information, often available in applications, to multivariate statistical procedures through order restrictions is receiving increasing attention during the last years as it allows to improve the performance of the procedures. Good examples of this trend are the papers by Rueda *et al.* (2009), Fernández *et al.* (2012) and Barragán *et al.* (2013) where this information is used to improve statistical procedures for circular data applied to cell biology, ElBarmi *et al.* (2010) where the information is used for estimating cumulative incidence functions in survival analysis, Ghosh *et al.* (2008) where it is used to make inferences on tumor size distributions or Davidov and Peddada (2013) where a test for multivariate stochastic order is applied to dose-response studies.

In this work, we deal with the incorporation of additional information to discriminant rules. Discriminant analysis is a well-known technique, first established by Fisher (1936), used in many science fields to define rules that allow to classify samples into a small number of populations based on a sample of observations whose population is known, usually called training set. To our best knowledge, the first paper considering additional information under the usual equal covariances assumption, which leads to linear discriminant rules, is Long and Gupta (1998). However, that paper provided limited results for the case of two populations with simple order restrictions and identity covariances matrices only. In a series of papers, the rules appearing in that initial paper have been improved, first to deal with more general types of information expressed in terms of cones of restrictions and general covariance matrices in Fernández *et al.* (2006) and later to the case of more than two populations in Conde *et al.* (2012). The robustness of the rules has also been studied in Salvador *et al.* (2008) and good estimators of the performance of the rules (which is an essential issue in discriminant analysis) have been provided in Conde *et al.* (2013). From now on, we will also refer to these rules as restricted rules as the additional information is incorporated through restrictions on the populations means.

The purpose of the present paper is two fold. The first is to introduce the **dawai** package, programmed in R environment, which can be downloaded from `http://cran.r-project.org/web/packages/dawai/`. This package provides all the functions needed to take advantage of the rules that incorporate additional information. The functions in the package allow to define the restricted rules, to classify the samples and to evaluate the accuracy of the results. The second contribution of this paper is the extension of the ideas given in previous papers from the case of equal covariances in the different populations to the case of unequal covariances among the populations and consequently the definition of the corresponding restricted quadratic discriminant rules, and also the definition of estimators of the accuracy of the rules for the general case where more than two populations appear in the problem.

In Section 2 we describe the statistical problem and the methodology of Fernández *et al.* (2006), Conde *et al.* (2012) and Conde *et al.* (2013) which we extend to the above mentioned situations. In Section 3 we introduce the **dawai** package and detail all the functions that it includes. The wide range of applications of the **dawai** package is illustrated in Section 4 using two data sets coming from two different fields such as biology and pattern recognition. Some concluding remarks are provided in Section 5.

2. Discriminant Analysis With Additional Information

We consider a finite number $k \geq 2$, of distinct populations of items $\Pi_1, \ldots, \Pi_k$. Each item is assumed to belong to one and only one of the populations. Let Z be a categorical variable identifying the population and let $X = (X_1, \ldots, X_p)'$ be the p-dimensional vector of predictors. Denote also as P_{XZ} the joint distribution of (X, Z), and as P_j the distribution of X in population Π_j with density function f_j, $j = 1, \ldots, k$. The classical discrimination problem deals with the classification of an observation $U = (U_1, \ldots, U_p)$, whose origin is unknown, into one of those populations. If we consider a 0-1 loss function and a priori probability π_j for the population Π_j, $j = 1, \ldots, k$, it is well known that the optimal classification rule, also called Bayes rule, is given by:

$$\text{Classify } U \text{ in } \Pi_j \text{ iff } \pi_j f_j(U) \geq \pi_l f_l(U),\ l = 1, 2, \ldots, k.$$

In applications, the density functions $f_j, j = 1, \ldots, k$, are unknown although there is sample information available. This sample information is contained in the so-called training sample given by a set of items for which both the predictors values and the correct population they belong to are registered. We represent the training sample as $M_n = \{(Y_i, Z_i), i = 1, \ldots, n\}$, where n is the items sample size, Y_i is the value that vector X takes at the i-th item in the sample and Z_i is the population the i-th item belongs to. Then, a classification rule is an application $R_n : \{\mathbb{R}^p \times \{1, \ldots, k\}\}^n \times \mathbb{R}^p \to \{1, \ldots, k\}$, that assigns a new observation $U \in \mathbb{R}^p$ for which the population is unknown to one of the k populations, $R_n(M_n, U) \in \{1, \ldots, k\}$.

From now on, we assume that $\pi_j = \frac{1}{k}$, $j = 1, \ldots, k$ (the case of unequal a priori probabilities is a trivial extension). If we further assume that $P_j \sim N_p(\mu_j, \Sigma)$, $j = 1, \ldots, k$, the optimal classification rule (the one with lowest expected loss) may be written as:

$$\text{Classify } U \text{ in } \Pi_j \text{ iff } (U - \mu_j)'\Sigma^{-1}(U - \mu_j) \leq (U - \mu_l)'\Sigma^{-1}(U - \mu_l), l = 1, \ldots, k.$$

Unfortunately, this rule cannot be used in practice as the mean vectors $\mu_j = (\mu_{j1}, \ldots, \mu_{jp})'$, $j = 1, \ldots, k$, and the common covariance matrix Σ are unknown. However, as we have a training sample these parameters may be estimated using respectively the sample vectors means $\overline{Y}_j$ and the pooled sample covariance matrix S,

$$\overline{Y}_j = (\overline{Y}_{j1}, \ldots, \overline{Y}_{jp})' = \frac{1}{n_j} \sum_{l=1}^{n} Y_l I_{(Z_l = j)} \quad \text{and}$$

$$S = \frac{1}{n-k} \sum_{j=1}^{k} \sum_{l=1}^{n} \left(Y_l - \overline{Y}_j\right) \left(Y_l - \overline{Y}_j\right)' I_{(Z_l = j)},$$

where $n_j = \sum_{l=1}^{n} I_{(Z_l = j)}$ is the sample size of population Π_j, $j = 1, \ldots, k$, and $n = \sum_{j=1}^{k} n_j$. As this estimated rule, obtained plugging the estimators in the initial rule, is linear in the predictors, it is usually known as linear discriminant rule or Fisher's rule:

$$\text{Classify } U \text{ in } \Pi_j \text{ iff } (U - \overline{Y}_j)'S^{-1}(U - \overline{Y}_j) \leq (U - \overline{Y}_l)'S^{-1}(U - \overline{Y}_l), l = 1, \ldots, k. \tag{1}$$

If the covariance matrices are not assumed to be equal, i. e., $P_j \sim N_p(\mu_j, \Sigma_j)$, $j = 1, \ldots, k$, the optimal rule can be written as:

$$\begin{aligned} \text{Classify } U \text{ in } \Pi_j \text{ iff } & -\frac{1}{2}\log(|\Sigma_j|) - \frac{1}{2}\left\{(U - \mu_j)'\Sigma_j^{-1}(U - \mu_j)\right\} \geq \\ & -\frac{1}{2}\log(|\Sigma_l|) - \frac{1}{2}\left\{(U - \mu_l)'\Sigma_l^{-1}(U - \mu_l)\right\}, \ l = 1, \ldots, k. \end{aligned}$$

Again, if we replace in this rule the unknown parameters μ_j and Σ_j by their corresponding estimators $\overline{Y}_j$ and $S_j = \frac{1}{n_j - 1} \sum_{l=1}^{n} \left(Y_l - \overline{Y}_j\right) \left(Y_l - \overline{Y}_j\right)' I_{(Z_l = j)}$, $j = 1, \ldots, k$, we obtain a rule that depends on the predictor in a quadratic way and it is therefore known as the restricted quadratic discriminant rule:

$$\begin{aligned} \text{Classify } U \text{ in } \Pi_j \text{ iff } & -\frac{1}{2}\log(|S_j|) - \frac{1}{2}\left\{(U - \overline{Y}_j)'S_j^{-1}(U - \overline{Y}_j), \right\} \geq \\ & -\frac{1}{2}\log(|S_l|) - \frac{1}{2}\left\{(U - \overline{Y}_l)'S_l^{-1}(U - \overline{Y}_l), \right\}, \ l = 1, \ldots, k. \end{aligned} \tag{2}$$

2.1. Restricted discriminant rules

In the introduction we referred applications where it is usual that some additional information is available. In many of these cases the information can be written as inequality restrictions among the population means. In the literature these restrictions are usually represented by a polyhedric cone (cf. Robertson *et al.* (1988) or Silvapulle and Sen (2005)). In our case, our pk-dimensional populations means will belong to a cone C in $\mathbb{R}^{pk}$,

$$(\mu_1', \ldots, \mu_k')' \in C = \left\{ x \in \mathbb{R}^{pk} : a_j' x \geq 0,\ j = 1, \ldots, m \right\}, \tag{3}$$

where the m vectors $a_j \in \mathbb{R}^{pk},\ j = 1, \ldots, m$, are determined by the restrictions imposed on the means.

Polyhedral cones are widely used in restricted inference literature, because they cover the most interesting cases from a practical standpoint. Among these cones, those representing order relations among the means are especially interesting. For example, it is not unusual to know that the observations from one of the populations, for example, Π_1 (which may be the control population in a medical study), take, in mean, lower values that those coming from any of the other populations for a subset $L \subseteq \{1, \ldots, p\}$ of predictor variables. In the usual restricted statistical terminology, we can say that there is a "tree order" among the populations means in the variables in L. In this case, we can write

$$(\mu_1', \ldots, \mu_k')' \in C_{TO} = \left\{ x \in \mathbb{R}^{pk} : x_l \leq x_{l+rp},\ r = 1, \ldots, k-1,\ l \in L \right\}. \tag{4}$$

Another usual situation is when it is known that there is an increase in the means of a subset L of predictors (for example, due to increased severity level in an illness study). This is known as a "simple order" among the populations means in the variables in L, and may be represented in R^{pk} using the cone

$$(\mu_1', \ldots, \mu_k')' \in C_{SO} = \left\{ x \in \mathbb{R}^{pk} : x_l \leq x_{l+p} \leq \cdots \leq x_{l+(k-1)p},\ l \in L \right\}. \tag{5}$$

Restricted linear discriminant rules

As mentioned above, in this case we assume $\Sigma_j = \Sigma,\ j = 1, \ldots, k$. Fernández *et al.* (2006) deal with this situation when the number of populations is $k = 2$. They propose a family of classification rules whose expected loss (total probability of misclassification) is lower than that of the linear discriminant rule (1). These rules are based on the use of additional information to obtain alternative estimators of the vector means. The generalization to the $k > 2$ populations case appears in Conde *et al.* (2012). These alternative estimators are defined via an iterative procedure whose convergence is shown in Fernández *et al.* (2006) and that is described here for completeness.

Consider the pk square matrix $S_*^{-1} = \left[diag\left(\frac{S}{n_1}, \frac{S}{n_2}, \ldots, \frac{S}{n_k} \right) \right]^{-1}$.

Definition 1 (Conde *et al.* (2012))

For $\gamma \in [0, 1]$, let $\widehat{\mu}^{\gamma}$ be the limit value, when $m \longrightarrow \infty$, of the following iterative procedure:

$$\widehat{\mu}^{\gamma(m)} = P_{S_*^{-1}}\left(\widehat{\mu}^{\gamma(m-1)} | C \right) - \gamma P_{S_*^{-1}}\left(\widehat{\mu}^{\gamma(m-1)} | C^P \right),\ m = 1, 2, \ldots,$$

where $\widehat{\mu}^{\gamma(0)} = \left(\overline{Y}_1^{'}, \ldots, \overline{Y}_k^{'}\right)^{'} \in \mathbb{R}^{pk}$, $P_{S_*^{-1}}(Y|C)$, *is the projection of* $Y \in \mathbb{R}^{pk}$ *onto the cone* C *using the metric given by the matrix* S_*^{-1}, *and* $C^P = \{y \in \mathbb{R}^{pk} : y' S_*^{-1} x \leq 0,\ x \in C\}$ *is the polar cone of* C.

The computation of the projection of a vector onto a polyhedral cone can be carried out using the `lsConstrain.fit` method contained in **ibdreg** R package. Figure 1 shows, in $\mathbb{R}^2$, the cones C and C^P and the estimators defined when $\gamma = 1$, exposing the need for an iterative procedure when C is an acute cone.

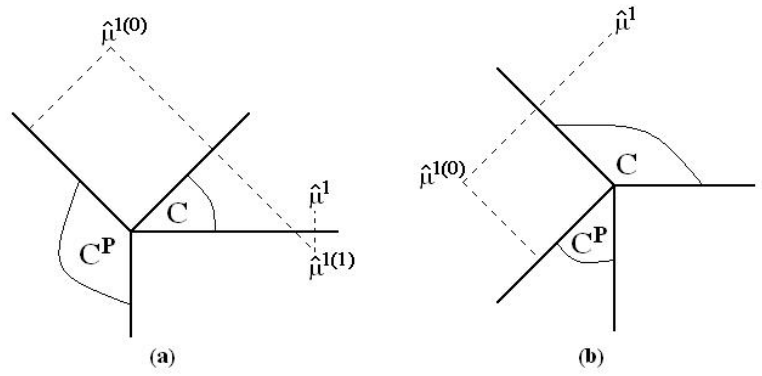

Figure 1: Examples of the iterative procedure for mean vector estimation for an acute (a) and a non-acute cone (b)

These estimators are plugged into the original rule to obtain the restricted lineal discriminant rules $R_l(\gamma)$:

$$\text{Classify } U \text{ in } \Pi_j \text{ iff } (U - \widehat{\mu}_j^{\gamma})' S_*^{-1} (U - \widehat{\mu}_j^{\gamma}) \leq (U - \widehat{\mu}_l^{\gamma})' S_*^{-1} (U - \widehat{\mu}_l^{\gamma}), l = 1, \ldots, k.$$

for $\gamma \in [0, 1]$.

For more details on these restricted linear rules and their properties the reader is referred to Fernández *et al.* (2006) and Conde *et al.* (2012).

Restricted quadratic discriminant rules

In many applications the covariance matrices of the different populations cannot be assumed to be equal. In this case for each $\gamma \in [0, 1]$ the estimator $\widehat{\mu}^{\gamma} = (\widehat{\mu}_1^{\gamma'}, \ldots, \widehat{\mu}_k^{\gamma'})'$ of $\mu = (\mu_1', \ldots, \mu_k')' \in C$ is obtained using the iterative procedure described in Definition 1, replacing the matrix S_*^{-1} by $\left[diag(\frac{S_1}{n_1}, \ldots, \frac{S_k}{n_k})\right]^{-1}$. Again, these estimators are plugged into the original rule to obtain the restricted quadratic discriminant rules: $R_q(\gamma)$

$$\begin{aligned} \text{Classify } U \text{ in } \Pi_j \text{ iff } & -\frac{1}{2}\log|S_j| - \frac{1}{2}\left\{(U - \widehat{\mu}_j^{\gamma})' S_j^{-1} (U - \widehat{\mu}_j^{\gamma}), \right\} \geq \\ & -\frac{1}{2}\log|S_l| - \frac{1}{2}\left\{(U - \widehat{\mu}_l^{\gamma})' S_l^{-1} (U - \widehat{\mu}_l^{\gamma}), \right\},\ l = 1, \ldots, k, \end{aligned}$$

for $\gamma \in [0, 1]$.

The restricted linear and quadratic discriminant rules can be defined with `rlda` and `rqda` functions of the R package **dawai**, respectively. The corresponding functions that allow to classify samples are `predict.rlda` and `predict.rqda`, respectively.

2.2. True error rate estimation

From the applications point of view, the evaluation of the classification rule for a given training sample is even more important than the expected loss of the rule. The true error rate, E_n, of the rule R_n, is the probability of misclassification of the rule given the training sample, i. e., $E_n = \mathsf{P}_{XZ}(R_n(M_n, U) \neq Z \mid M_n)$. It is well known that the best way of estimating the true classification error of a classification rule is the use of an independent sample, usually called test sample. However, in practice it is common that the sample size is not large enough to split the sample into a training and a test sample as that would decrease the efficiency of the rule. For this reason, the estimation of E_n for the usual rules such as for example Fisher's linear rule (1), the quadratic discriminant rule (2), the nearest neighbors rules (Cover and Hart 1967) or random forest rules (Breiman 2001), is a widely studied topic in the literature. Parametric and non-parametric estimators of E_n have been proposed and non-parametric estimators based on resampling have shown a good performance for the above mentioned rules. Schiavo and Hand (2000) summarizes the work made on this topic until that date. More recent references are, for instance, Steele and Patterson (2000), Wehberg and Schumacher (2004), Fu *et al.* (2005), Molinaro *et al.* (2005), Kim and Cha (2006), Kim (2009) or Borra and Di Ciaccio (2010).

Conde *et al.* (2013) propose four new estimators of E_n specific for the restricted linear discriminant rule $R_l(\gamma)$ for $k = 2$ populations. Two of them, $BT2$ and $BT3$, are generated from the leave-one-out bootstrap ($LOOBT$, see Efron (1983)). The other two, $BT2CV$ and $BT3CV$, are cross-validation after bootstrap (BCV, see Fu *et al.* (2005)) versions of $BT2$ and $BT3$ respectively. In the following, we describe the generalization of these estimators to the $k > 2$ populations and to the restricted quadratic discrimination cases. The underlying idea in the definition of the new estimators of the true error rate is that the "bootstrap world" should mirror the "real world". We present two proposals: the first one is to modify the restrictions cone, the second one is to adapt the training sample.

The BT2 and BT2CV estimators

Assume that the additional information is written as in (3). Let us denote as $\overline{C}$ the following random cone generated by the sample mean vectors $\overline{Y} = \left(\overline{Y}_1', \ldots, \overline{Y}_k'\right)'$:

$$\overline{C} = \left\{ x \in \mathbb{R}^{pk} \begin{array}{lll} a_j' x \geq 0 & \text{if} & a_j' \overline{Y} \geq 0 \\ a_j' x \leq 0 & \text{if} & a_j' \overline{Y} < 0 \end{array}, \; j = 1, \ldots, m \right\},$$

i. e., the cone determined by the restrictions verified by the sample means.

The true error rate estimator $BT2$ of the restricted linear or quadratic classification rules ($R_l(\gamma)$, $R_q(\gamma)$) is computed in a way similar to $LOOBT$ but considering bootstrap classification rules generated using projections onto cone $\overline{C}$ instead of C for each bootstrap training sample. $LOOBT$ is the porportion of wrongly classified observations using the following procedure: B bootstrap samples are considered and the corresponding B bootstrap versions of the classification rule are used for classifying the original observations that do not belong to the bootstrap training sample. A bootstrap training sample $M_n^* = \{(Y_i^*, Z_i^*), i = 1, \ldots, n\}$

is a size n randomly obtained (with replacement) sample from the original training sample (i. e. $\mathsf{P}((Y_i^*, Z_i^*) = (Y_s, Z_s)) = \frac{1}{n}$ with $s, i \in \{1, \ldots, n\}$). B such bootstrap samples $M_n^{*b} = \{(Y_i^{*b}, Z_i^{*b}), i = 1, \ldots, n\}$, $b = 1, \ldots, B$, are obtained from M_n. For each bootstrap training sample we define the bootstrap version of the estimator of $\mu = (\mu_1', \ldots, \mu_k')'$ that we denote as μ_γ^{*b} (with $\gamma \in [0, 1]$), as the limit when $i \to \infty$ of the following iterative procedure similar to the one considered in Definition 1. Let $\widehat{\mu}_\gamma^{(0)b} = \overline{Y}$ and

$$\widehat{\mu}_\gamma^{(i)b} = P_A\left(\widehat{\mu}_\gamma^{(i-1)b} | \overline{C}\right) - \gamma P_A\left(\widehat{\mu}_\gamma^{(i-1)b} | \overline{C}^P\right), \ i = 1, 2, \ldots,$$

where matrix A is equal to $\left[diag(\frac{S}{n_1}, \frac{S}{n_2}, \ldots, \frac{S}{n_k})\right]^{-1}$ for the restricted linear discriminant rule and equal to $\left[diag(\frac{S_1}{n_1}, \frac{S_1}{n_2}, \ldots, \frac{S_k}{n_k})\right]^{-1}$ for the restricted quadratic discriminant rule.

Now, we denote as $R_l^{*b}(\gamma)$ and $R_q^{*b}(\gamma)$ the bootstrap versions of the classification rules $R_l(\gamma)$ and $R_q(\gamma)$, respectively. For each of the B bootstrap rules we classify the observations in the original training sample that do not belong to the corresponding bootstrap sample M_n^{*b}. The true error rate estimator $BT2$ is the proportion of observations wrongly classified.

The $BT2CV$ estimator is the BCV (Fu *et al.* 2005) version of $BT2$. For each of the B bootstrap training samples, let CV_b be the true error rate estimator obtained using the cross-validation method on sample M_n^{*b}. Then $BT2CV = \frac{1}{B}\sum_{b=1}^{B} CV_b$.

The BT3 and BT3CV estimators

The true error rate estimator denoted as $BT3$ is based in adapting the original training sample, instead of modifying the cone C like in $BT2$, as follows.

Assume that the original training sample $M_n = \{(Y_i, Z_i), i = 1, \ldots, n\}$ does not verify the restrictions, i. e., $\overline{Y} \notin C$. For any $\gamma \in [0, 1]$, let $\widehat{\mu}_j^\gamma$ be the restricted estimator of μ_j obtained in Definition 1. Now, we transform the original training sample in such a way that the new sample means belong to C. The transformed training sample is $\{(W_i, Z_i), i = 1, \ldots, n\}$, where

$$W_i = Y_i - \overline{Y}_j + \widehat{\mu}_j^\gamma \text{ if } Z_i = j,$$

for $i = 1, \ldots, n$ and $j = 1, \ldots, k$.

In this way $\overline{W} = (\overline{W}_1', \ldots, \overline{W}_k')'$, $\overline{W}_j = \frac{1}{n_j}\sum_{l=1}^{n} W_l I_{(Z_l = j)}$, $j = 1, \ldots, k$. Now, the estimator denoted as $BT3$ is computed in a similar way to that of $LOOBT$ but replacing the original training sample by the transformed one.

The $BT3CV$ estimator is the cross-validation after bootstrap version of $BT3$.

These four estimators of the true error rates of the restricted linear and quadratic discriminant rules can be obtained with `err.est.rlda` and `err.est.rqda` functions of the R package **dawai**, respectively.

3. Package dawai

The R package **dawai** consists of a total of six functions, three for each of the two restricted discrimination analysis situations (i. e., equal or unequal covariances in the populations). The three functions for each case are: one to define the rules that take into account the additional information expressed in terms of restrictions on the populations means and classify the samples in the training set; a second one which predicts the populations of new samples using the previously defined rule; and, finally, a third one which can evaluate the accuracy of the rules associated to the training set.

We start this section giving some background on R packages for performing discriminant analysis. We then explain the functions of this package.

3.1. Related packages

As discriminant analysis is a well-known and widely used technique there are many packages in R for performing discriminant analysis. The basic procedures are in package

- **MASS** (Ripley *et al.* 2011): Support Functions and Datasets for Venables and Ripley's MASS.

Some more recent packages including new features and discrimination in specific conditions are

- **mda** (Hastie *et al.* 2013): Mixture and flexible discriminant analysis.
- **rrlda** (Gschwandtner *et al.* 2013): Robust Regularized Linear Discriminant Analysis.
- **sparsediscrim** (Ramey 2013): Sparse Discriminant Analysis.

Since none of the existing packages for discriminant analysis are applicable for performing discriminant analysis under restrictions, in this article we introduce the package "*d*iscriminant *a*nalysis *w*ith *a*dditional *i*nformation", with the acronym **dawai**.

Our package depends on **boot** (Ripley 2013) for bootstrapping, **ibdreg** (Sinnwell and Schaid 2013) for computing the projection of a vector onto a polyhedral cone with `lsConstrain.fit`, and **mvtnorm** (Genz *et al.* 2013) for computing multivariate normal densities. These packages should be installed before loading **dawai**.

3.2. The `rlda`, `predict.rlda` and `err.est.rlda` functions

As aforementioned, there are three functions with regard to the described restricted linear discriminant rules in the **dawai** package.

Firstly, the `rlda` function builds restricted linear classification rules with additional information expressed as inequality restrictions on the populations means, using the methodology developed in Fernández *et al.* (2006) and Conde *et al.* (2012).

The `rlda` function creates an S3 object of class `"rlda"`, explained below. The arguments of this function are summarized in Table 1.

Arguments	Values
`formula`	formula specifying the grouping factor and the variables
`data`	data frame from which variables in `formula` are to be taken
`x`	(if no `formula` given) data frame or matrix with the explanatory variables
`grouping`	(if no `formula` given) numeric vector or numeric levels factor with the classes of the observations
`subset`	index vector specifying the cases to be used in the training sample
`resmatrix`	matrix specifying the linear restrictions on the mean vectors
`resvector`	vector of constant values. If unspecified, a vector of 0's is used
`restext`	(if no `resmatrix` given) character string from which `resmatrix` is created
`gamma`	vector with the gamma values. If unspecified, `c(0,1)` is used
`prior`	prior probabilities of class membership. If unspecified, the class proportions for the training set are used

Table 1: Arguments of the `rlda` function.

The `rlda` function may be called giving either a formula and data frame, or a data frame and grouping factor, or a matrix and grouping factor, as the first two arguments. Arguments `resmatrix` or `restext` must be specified. All other arguments are optional.

Classes of the observations must be identified, either in a colum of `data` or in the `grouping` vector, by natural numbers varying from 1 to the number of classes, which must be greater than 1.

Arguments `resmatrix` and `resvector` determine the additional information on the means vector: `resmatrix` $(\mu_1', \ldots, \mu_k')' \leq$ `resvector`.

The purpose of `restext` is to make easier the specification of the two most usual cones of restrictions, the tree order (4) and the simple order (5) cones. The first element of `restext` must be either "s" (simple order) or "t" (tree order), the second element must be either "<" (increasing componentwise order) or ">" (decreasing componentwise order), and the rest of the elements should be numbers from 1 to the number of explanatory variables, separated by commas, specifying among which variables the restrictions hold.

If there are missing values in either `data`, `x` or `grouping`, the corresponding observations will be deleted.

The `rlda` function creates an S3 object of class `"rlda"`, which is a list with the following elements:

`$call` is the (matched) function call.

`$trainset` is a matrix with the training set used (first columns) and the class for each observation (last column).

`$restrictions` is an edited character string with the linear restrictions on the mean vectors detailed.

`$resmatrix` is the matrix with the restrictions on the mean vectors used.

`$resvector` is the constant values vector used.

`$prior` is a vector containing the prior probabilities of class membership used.

`$counts` is the number of observations of each of the classes used.

`$N` is the total number of observations used.

`$samplemeans` is a matrix with the sample means in rows.

`$samplevariances` is an array with the sample covariance matrices of the classes.

`$gamma` are the gamma values used.

`$spooled` is the pooled covariance matrix.

`$estimatedmeans` is an array with the estimated means for each classification rule.

`$apparent` is the apparent error rate for each classification rule.

The `print.rlda(x, ...)` method is an S3 method defined to print an object `x` of class `"rlda"`.

The second function `predict.rlda` matches the parameters of the generic function `predict` and adds some more. The arguments of this function are summarized in Table 2. This function classifies multivariate observations contained in a data frame `newdata` using the restricted linear classification rules defined in an `object` of class `"rlda"`. All other arguments are optional.

Arguments	Values
`object`	object of class `"rlda"`
`newdata`	data frame of cases to be classified. It must contain the variables in `object`
`prior`	prior probabilities of class membership. If unspecified, `object$prior` is used
`gamma`	gamma values. If specified, its values must be contained in `object$gamma`. If unspecified, `object$gamma` is used
`grouping`	numeric vector or numeric levels factor with the classes of the observations in `newdata`. If present, true error rate will be estimated from `newdata`

Table 2: Arguments of the `predict.rlda` function.

The output of this function is a list with the following elements:

`$call` is the (matched) function call.

`$class` is a matrix with the classification for each rule (in columns).

`$prior` is a vector containing the prior probabilities of the classes used.

`$posterior` is an array with the posterior probabilities for each rule.

`$error.rate` is the true error rate estimation (when grouping is specified) for each rule based on `newdata`.

Finally, the third function `err.est.rlda` estimates the true error rate of the restricted linear classification rules defined in an object `x` of class `"rlda"`, using the methodology developed in Conde *et al.* (2013) and in Section 2.2. of this paper. The arguments of this method are summarized in Table 3.

Arguments	Values
`x`	object of class `"rlda"`
`nboot`	number of bootstrap samples used to estimate the true error rate. If unspecified, `nboot` = 50
`gamma`	gamma values. If specified, its values must be contained in `x$gamma`. If unspecified, `x$gamma` is used
`prior`	prior probabilities of class membership. If unspecified, `x$prior` is used

Table 3: Arguments of the `err.est` method.

The output of the `err.est.rlda` function is a list with the following elements:

`$call` is the (matched) function call.

`$restrictions` is a character vector with the restrictions on the means vector detailed.

`$prior` is the prior probabilities of the classes used.

`$counts` is the number of observations of the classes used.

`$N` is the total number of observations used.

`$estimationError` is a matrix with $BT2$, $BT3$, $BT2CV$ and $BT3CV$ true error rate estimates of the rules.

Examples to illustrate these functions are provided in Section 4.1.

3.3. The `rqda`, `predict.rqda` and `err.est.rqda` functions

The `rqda`, `predict.rqda` and `err.est.rqda` functions are the corresponding versions of the `rqda`, `predict.rqda` and `err.est.rqda` functions for performing restricted quadratic discrimination.

The `rqda` function builds restricted quadratic classification rules using the methodology developed in Section 2.1. The arguments of the `rqda` function are the same arguments of the `rlda` function, summarized in Table 1. The `rqda` function creates an S3 object of class `"rqda"`, which is a list with the same elements as class `"rlda"` except `spooled`, the pooled covariance matrix, not used in quadratic classification.

The method `print.rqda(x, ...)` is an S3 method defined to print an object `x` of class `"rlda"`.

The `predict.rqda` function classifies multivariate observations with restricted quadratic classification rules. The arguments of this function are the same as `predict.rlda`, summarized in Table 2. This function classifies a data frame `newdata` with restricted quadratic classification

rules defined in an `object` of class `"rqda"`. The output of this function is a list with the same elements as the output of the `predict.rlda` function.

Finally, the `err.est.rqda` function estimates the true error rate of restricted quadratic classification rules using the methodology developed in Section 2.2. The arguments of this function are the same as the arguments of the `err.est.rlda` function, summarized in Table 3, where `x` is an object of class `"rqda"`.

Examples to illustrate these functions are provided in Section 4.2.

4. Applications

There is a wide range of applications of the **dawai** package, which we illustrate in this section using two data sets coming from two different fields such as biology and pattern recognition.

4.1. Biological application

In patient care, as for example in cancer treatment, an important component is the correct classification of the patient into one of the disease stages. The disease stages correspond to increasingly advanced levels of the disease, so it is reasonable to expect the mean values of some variables to increase or decrease with the severity of the illness. This is the case of primary biliary cirrhosis (PBC), an autoimmune liver disease causing liver inflammation and a gradual destruction of the intrahepatic bile ducts found within the liver. PBC is a progressive disease, with four sucessive stages as time passes (Scheuer 1967).

The data set we will use now, called `pbc`, is in **survival** R package, taken from Therneau and Grambsch (2000). This data set is from the Mayo Clinic trial in PBC of the liver conducted between 1974 and 1984, and it has 418 cases and 20 variables.

We will use this data set to exemplify the restricted linear discriminant rules. We consider three variables as predictors ($p = 3$), *Bili*, *Albumin* and *Platelet* (the amounts of serum bilirunbin (mg/dl) and serum albumin (g/dl) and platelet count, respectively), and three populations ($k = 3$), joining the original stages 1 and 2 into one so that the classes have enough elements to split the sample into training and a test data sets of reasonable size, as seen below.

```
R> library(survival)
R> data("pbc")
R> data <- pbc[, c("bili", "albumin", "platelet", "stage")]
```

Let us take complete observations only.

```
R> data <- data[rowSums(is.na(data)) == 0, ]
```

We transform logarithmically the values of the explicative variables so that the variables are approximately normally distributed.

```
R> data <- cbind(data[, "stage", drop = FALSE],
+                        "logBili" = log(data[["bili"]]),
+                        "logAlbumin" = log(data[["albumin"]]),
+                        "logPlatelet" = log(data[["platelet"]]))
R> data$stage <- as.factor(data$stage)
R> levels(data$stage)

[1] "1" "2" "3" "4"

R> table(data$stage)

  1   2   3   4
 20  86 153 142
```

These are the number of elements in each of the four classes. Notice that there is a low number in the first class and that 401 cases out of the 418 initial ones have no missing values in the three predictor variables. As mentioned before, we join stages "1" and "2" and relabel them.

```
R> levels(data$stage) <- c(1, 1, 2, 3)
R> table(data$stage)

  1   2   3
106 153 142
```

We will consider the restrictions between population means given by the whole data set: $\mu_{1,1} \leq \mu_{2,1} \leq \mu_{3,1}$, $\mu_{1,2} \geq \mu_{2,2} \geq \mu_{3,2}$, $\mu_{1,3} \geq \mu_{2,3} \geq \mu_{3,3}$, i. e., the amount of serum bilirunbin increases and the amount of serum albumin and platelet count decrease with PBC stage. To establish these restrictions, we define a restrictions matrix (`resmatrix`).

```
R> A <- matrix(0, ncol = 9, nrow = 6)
R> A[t(matrix(c(1, 1, 4, 4, 2, 5, 3, 6, 5, 8, 6, 9), nrow = 2))] <- 1
R> A[t(matrix(c(1, 4, 4, 7, 2, 2, 3, 3, 5, 5, 6, 6), nrow = 2))] <- -1
R> A

     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
[1,]    1    0    0   -1    0    0    0    0    0
[2,]    0   -1    0    0    1    0    0    0    0
[3,]    0    0   -1    0    0    1    0    0    0
[4,]    0    0    0    1    0    0   -1    0    0
[5,]    0    0    0    0   -1    0    0    1    0
[6,]    0    0    0    0    0   -1    0    0    1
```

As the restrictions are expressed as `resmatrix` $(\mu_1', \mu_2', \mu_3')' \leq$ `resvector`, the parameter `resvector` must be $(0, 0, 0, 0, 0, 0)'$, so we will not need to specify it (see Table 1).

We split the data set into a randomly selected training set and test set, fixing a seed in order to get the same results as the reader.

```
R> set.seed(-13615)
R> values <- runif(dim(data)[1])
R> trainsubset <- (values < 0.25)
R> testsubset <- (values >= 0.25)
```

Now we can build the restricted linear discriminant rules. Let us consider equal a priori probabilities.

```
R> obj <- rlda(stage ~ logBili + logAlbumin + logPlatelet, data,
+              subset = trainsubset, gamma = c(0, 0.75, 1),
+              resmatrix = A, prior = c(1/3, 1/3, 1/3))
R> obj

Restrictions:
   mu1,1 - mu2,1 <= 0
 - mu1,2 + mu2,2 <= 0
 - mu1,3 + mu2,3 <= 0
   mu2,1 - mu3,1 <= 0
 - mu2,2 + mu3,2 <= 0
 - mu2,3 + mu3,3 <= 0

Prior probabilities of classes:
   class1    class2    class3
0.3333333 0.3333333 0.3333333

Apparent error rate (%):
   gamma=0 gamma=0.75    gamma=1
  50.00000   53.84615   54.80769
```

Now we consider the test set, containing the observations in `data` not present in the training set, and classify them. As we know the classes that the observations in the test set belong to, we can estimate the true error rate.

```
R> pred <- predict(obj, newdata = data[testsubset,],
+                  grouping = data[testsubset, "stage"])
R> pred$error.rate

                      gamma=0 gamma=0.75  gamma=1
True error rate (%): 54.20875   52.86195 51.51515
```

The fact that the apparent error rate increases and the true error rate decreases with γ is a typical effect for these restricted rules, see Fernández *et al.* (2006), Conde *et al.* (2012) and Conde *et al.* (2013).

Finally, we estimate the true error rate from the training sample with `nboot` $= 50$ (the default value) and $\gamma = 0.75, 1$.

```
R> err.est(obj, gamma = c(0.75, 1))
```

```
Restrictions:
   mu1,1 - mu2,1 <= 0
 - mu1,2 + mu2,2 <= 0
 - mu1,3 + mu2,3 <= 0
   mu2,1 - mu3,1 <= 0
 - mu2,2 + mu3,2 <= 0
 - mu2,3 + mu3,3 <= 0

Prior probabilities of classes:
   class1    class2    class3
0.3333333 0.3333333 0.3333333

True error rate estimation (%):
      gamma=0.75  gamma=1
BT2     53.89034 53.31593
BT3     55.45692 55.56136
BT2CV   50.23077 50.53846
BT3CV   51.90385 51.63462
```

4.2. Pattern recognition application

As an example of pattern recognition, we will use a data set contained in `dawai` package called `Vehicle2`.

```
R> data("Vehicle2")
```

This data set is a subset from the `Vehicle` data set, available in the R package **mlbench** and taken from the UCI Repository Of Machine Learning Databases (http://www.ics.uci.edu/~mlearn/MLRepository.html), originally gathered in Siebert (1987). The purpose of the data set is to study how to distinguish 3D objects from a 2D image, i. e., how to classify a given silhouette as viewed from a camera from different angles and elevations into one of four types of vehicle, using a set of features extracted from the silhouette. The vehicles used were a double-decker bus, a Cheverolet van, a Saab 9000 and an Opel Manta 400, with the expectation that the bus, the van and either one of the cars would be readily distinguishable, but it would be more difficult to distinguish between the cars.

`Vehicle2` is a data frame with 846 observations on 4 variables, all numerical and one nominal defining the class of the objects, the vehicle.

We will use this data set to exemplify the restricted quadratic discriminant rules.

We consider the three variables as predictors ($p = 3$) and the four available populations ($k = 4$).

```
R> data <- Vehicle2[, 1:3]
R> grouping <- Vehicle2$Class
R> levels(grouping)

[1] "bus"  "opel" "saab" "van"
```

```
R> levels(grouping) <- c(4, 1, 2, 3)
```

We have "ordered" the populations in terms of the vehicle size. The third variable, *Holl.Ra*, is the quotient (hollows area)/(bounding polygon area), and it could be reasonable to think that the means decrease with the vehicle size, so let us suppose the following restrictions on the means: $\mu_{13} \geq \mu_{23} \geq \mu_{33} \geq \mu_{43}$. We specify these restrictions by `restext = "s>3"`.

We split the data set into a randomly selected training set and test set, fixing a seed in order to get the same results as the reader.

```
R> set.seed(6550)
R> values <- runif(dim(data)[1])
R> trainsubset <- (values < 0.25)
```

Now we can build the restricted quadratic discriminant rules:

```
R> obj <- rqda(data, grouping, subset = trainsubset, restext = "s>3")
R> obj

Restrictions:
 - mu1,3 + mu2,3 <= 0
 - mu2,3 + mu3,3 <= 0
 - mu3,3 + mu4,3 <= 0

Prior probabilities of classes:
   class1    class2    class3    class4
0.2060302 0.2663317 0.2562814 0.2713568

Apparent error rate (%):
 gamma=0  gamma=1
55.77889 58.79397
```

Note that, as we have not specifed neither `gamma` nor `prior`, restricted rules are by default obtained for $\gamma = 0, 1$, and the class proportions of the training set are used as the prior probabilities of class membership.

Now we consider the test set, containing the observations in `data` not present in the training set, and classify them. As we know the classes of the observations in the test set, we can estimate the true error rate.

```
R> testsubset <- (values >= 0.25)
R> pred <- predict(obj, newdata = data[testsubset,],
+                  grouping = grouping[testsubset])
R> pred$error.rate

                      gamma=0 gamma=1
True error rate (%): 62.75116  61.051
```

We can also use equal a priori probabilities. We only need to specify them.

```
R> pred <- predict(obj, newdata = data[testsubset,],
+                  grouping = grouping[testsubset], prior = rep(1/4, 4))
R> pred$error.rate

                      gamma=0  gamma=1
True error rate (%): 61.36012 62.13292
```

Finally, we estimate the true error rate from the training sample, with `nboot` = 60 and equal a priori probabilities.

```
R> err.est(obj, nboot = 60, prior = rep(1/4,4))

Restrictions:
 - mu1,3 + mu2,3 <= 0
 - mu2,3 + mu3,3 <= 0
 - mu3,3 + mu4,3 <= 0

Prior probabilities of classes:
class1 class2 class3 class4
  0.25   0.25   0.25   0.25

True error rate estimation (%):
       gamma=0  gamma=1
BT2   62.82842 62.66849
BT3   63.05689 62.87411
BT2CV 56.64154 56.47404
BT3CV 57.06868 57.29481
```

5. Conclusions

In this paper the R package **dawai** has been presented. The package provides the functions needed to define linear or quadratic classification rules under order restrictions, to classify the samples and to evaluate the accuracy of the rules.

We have also extended in this paper the definitions given in previous papers (Fernández *et al.* (2006), Conde *et al.* (2012), Conde *et al.* (2013)) from the case of equal covariances in the different populations to the case of unequal covariances among the populations and consequently defined the corresponding restricted quadratic discriminant rules. Another novelty is the definition of estimators of the accuracy of the rules for the general more than two populations case, for restricted linear and quadratic discriminant rules, thus completing the procedures presented in those three previous papers.

Though we have illustrated the proposed methodology using examples from biology and pattern recognition, the software can obviously be applied to a wide range of contexts such as medical image analysis, drug discovery and development, optical character and handwriting

recognition, document classification, credit scoring... We expect the software described to be useful for researchers working in any of those fields.

Acknowledgments

Research partially supported by Spanish MCI grant MTM2012-37129.

References

Barragán S, Fernández M, Rueda C, Peddada S (2013). "isocir: An R Package for Constrained Inference using Isotonic Regression for Circular Data, with an Application to Cell Biology." *Journal of Statistical Software*, **54**(4), 1–17. URL http://www.jstatsoft.org/v54/i04.

Borra S, Di Ciaccio A (2010). "Measuring the prediction error. A comparison of cross-validation, bootstrap and covariance penalty methods." *Computational Statistics and Data Analysis*, **54**(12), 2976–2989.

Breiman L (2001). "Random forests." *Machine Learning*, **45**(1), 5–32.

Conde D, Fernández M, Rueda C, Salvador B (2012). "Classification of samples into two or more ordered populations with application to a cancer trial." *Statistics in Medicine*, **31**(28), 3773–3786.

Conde D, Salvador B, Rueda C, Fernández M (2013). "Performance and estimation of the true error rate of classification rules built with additional information. An application to a cancer trial." *Statistical Applications in Genetics and Molecular Biology*, **12**(5), 583–602.

Cover T, Hart P (1967). "Nearest neigbor pattern classification." *Information Theory, IEEE Transactions on*, **13**(1), 21–27.

Davidov O, Peddada S (2013). "Testing for the Multivariate Stochastic Order among Ordered Experimental Groups with Application to Dose-Response Studies." *Biometrics*, **101**(9), 1903–1909.

Efron B (1983). "Estimating the error rate of a prediction rule: Improvement on cross-validation." *Journal of the American Statistical Association*, **78**, 316–331.

ElBarmi H, Johnson M, Mukerjee H (2010). "Estimating cumulative incidence functions when the life distributions are constrained." *Journal of Multivariate Analysis*, **101**(9), 1903–1909.

Fernández M, Rueda C, Peddada S (2012). "Identification of a Core Set of Signature Cell Cycle Genes Whose Relative Order of Time to Peak Expression is Conserved Across Species." *Nucleic Acids Research*, **40**(7), 2823–2832. URL http://nar.oxfordjournals.org/content/40/7/2823.

Fernández M, Rueda C, Salvador B (2006). "Incorporating additional information to normal linear discriminant rules." *Journal of the American Statistical Association*, **101**(474), 569–577.

Fisher R (1936). "The use of multiple experiments in taxonomic problems." *Annals of Eugenics*, **7**, 179–188.

Fu W, Carroll R, Wang S (2005). "Estimating misclassification error with small samples via bootstrap cross-validation." *Bioinformatics*, **21**, 1979–1986.

Genz A, Bretz F, Miwa T, Mi X, Leisch F, Scheipl F, Bornkamp B, Hothorn T (2013). ***mvtnorm**: Multivariate Normal and t Distributions*. R package version 0.9-9996, URL http://cran.r-project.org/package=mvtnorm.

Ghosh D, Banerjee M, Biswas P (2008). "Inference for Constrained Estimation of Tumor Size Distributions." *Biometrics*, **64**, 1009–1017.

Gschwandtner M, Filzmoser P, Croux C, Haesbroeck G (2013). ***rrlda**: Robust Regularized Linear Discriminant Analysis*. R package version 1.1, URL http://cran.r-project.org/package=rrlda.

Hastie T, Tibshirani R, Leisch F, Hornik K, Ripley BD (2013). ***mda**: Mixture and flexible discriminant analysis*. R package version 0.4-4, URL http://cran.r-project.org/package=mda.

Kim J (2009). "Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap." *Computational Statistics and Data Analysis*, **53**(11), 3735–3745.

Kim J, Cha E (2006). "Estimating prediction errors in binary classification problem: Cross-validation versus bootstrap." *Korean Communications in Statistics*, **13**, 151–165.

Long T, Gupta R (1998). "Alternative linear classification rules under order restrictions." *Communications in Statistics - Theory and Methods*, **27**(3), 559–575.

Molinaro A, Simon R, Pfeiffer R (2005). "Prediction error estimation: a comparison of resampling methods." *Bioinformatics*, **15**, 3301–3307.

Ramey JA (2013). ***sparsediscrim**: Sparse Discriminant Analysis*. R package version 0.1.0, URL http://cran.r-project.org/package=sparsediscrim.

Ripley B (2013). ***boot**: Bootstrap Functions (originally by Angelo Canty for S)*. R package version 1.3-9, URL http://cran.r-project.org/package=boot.

Ripley B, Venables B, Bates DM, Hornik K, Gebhardt A, Firth D (2011). ***MASS**: Support Functions and Datasets for Venables and Ripley's MASS*. R package version 7.3-29, URL http://cran.r-project.org/package=MASS.

Robertson T, Wright F, Dykstra R (1988). *Order Restricted Statitical Inference*. John Wiley & Sons.

Rueda C, Fernández M, Peddada S (2009). "Estimation of Parameters Subject to Order Restrictions on a Circle with Application to Estimation of Phase Angles of Cell-Cycle Genes." *Journal of the American Statistical Association*, **104**(485), 338–347.

Salvador B, Fernández M, Martín I, Rueda C (2008). "Robustness of classification rules that incorporate additional information." *Computational Statistics and Data Analysis*, **52**(5), 2489–2495.

Scheuer P (1967). "Primary biliary cirrhosis." *Proceedings of the Royal Society of Medicine*, **60**(12), 1257–1260.

Schiavo R, Hand D (2000). "Ten More Years of Error Rate Research." *International Statistical Review*, **68**, 295–310.

Siebert JP (1987). "Vehicle Recognition Using Rule Based Methods." *Technical Report TIRM-87-018*, Turing Institute, Glasgow, Scotland.

Silvapulle MJ, Sen PK (2005). *Constrained Statistical Inference*. John Wiley & Sons.

Sinnwell JP, Schaid DJ (2013). ***ibdreg**: Regression Methods for IBD Linkage With Covariates*. R package version 0.2.-5, URL http://cran.r-project.org/package=ibdreg.

Steele B, Patterson D (2000). "Ideal bootstrap estimation of expected prediction error for k-nearest neighbor classifiers: Applications for classification and error assessment." *Statistics and Computing*, **10**(4), 349–355.

Therneau T, Grambsch P (2000). *Modeling Survival Data: Extending the Cox Model*. Springer-Verlag.

Wehberg S, Schumacher M (2004). "A comparison of nonparametric error rate estimation methods in classification problems." *Biometrical Journal*, **46**, 35–47.

Affiliation:

David Conde, Miguel A. Fernández, Bonifacio Salvador, Cristina Rueda
Departamento de Estadística e Investigación Operativa
Instituto de Matemáticas (IMUVA)
Universidad de Valladolid
Valladolid, Spain
E-mail: dconde@eio.uva.es,
miguelaf@eio.uva.es,
bosal@eio.uva.es,
crueda@eio.uva.es
URL: http://www.eio.uva.es/~david/
http://www.eio.uva.es/~miguel/
http://www.eio.uva.es/~boni/
http://www.eio.uva.es/~cristina/