Fiche TD avec le logiciel  : course5

Linear Discriminant Analysis

A.B. Dufour

Contents

1 Fisher's iris dataset

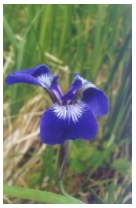
The data were collected by Anderson [1] and used by Fisher [2] to formulate the linear discriminant analysis (LDA or DA). The dataset gives the measurements in centimeters of the following variables: 1- sepal length, 2- sepal width, 3- petal length, and 4- petal width, this for 50 flowers from each of the 3 species of iris considered. The species considered are *Iris setosa*, *versicolor*, and *virginica*.

```
data(iris)
names(iris)
```

```
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"  "Species"
```

```
dim(iris)
```

```
[1] 150   5
```


setosa

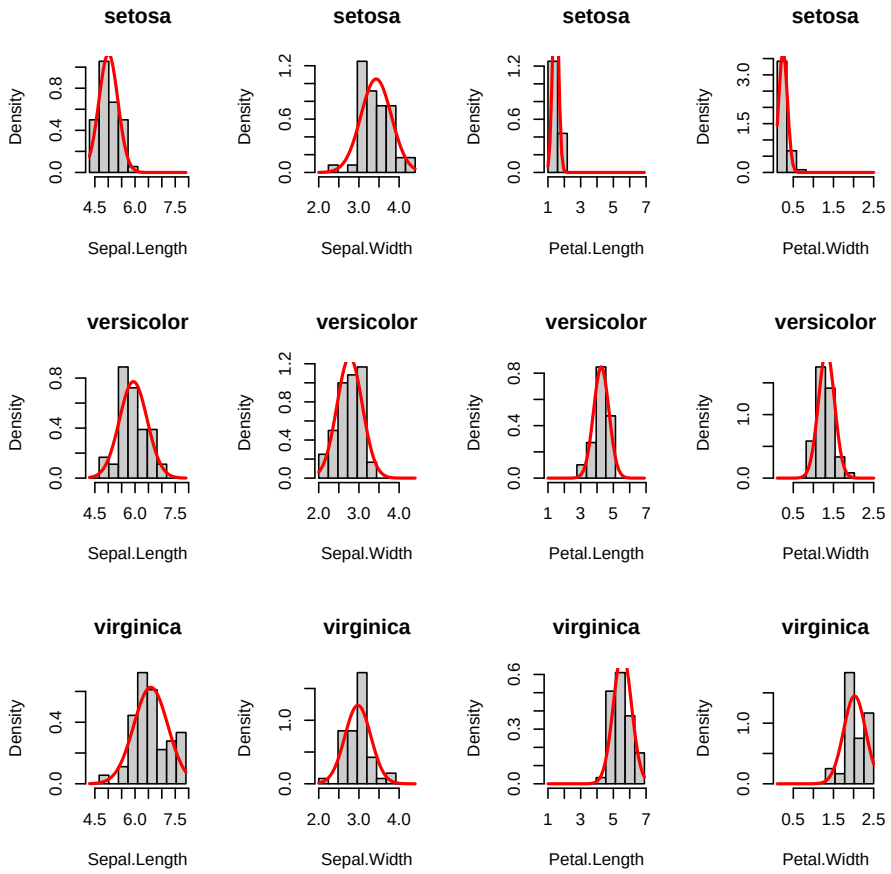

versicolor


virginica

[1]

One can build the histograms by species and by variable:

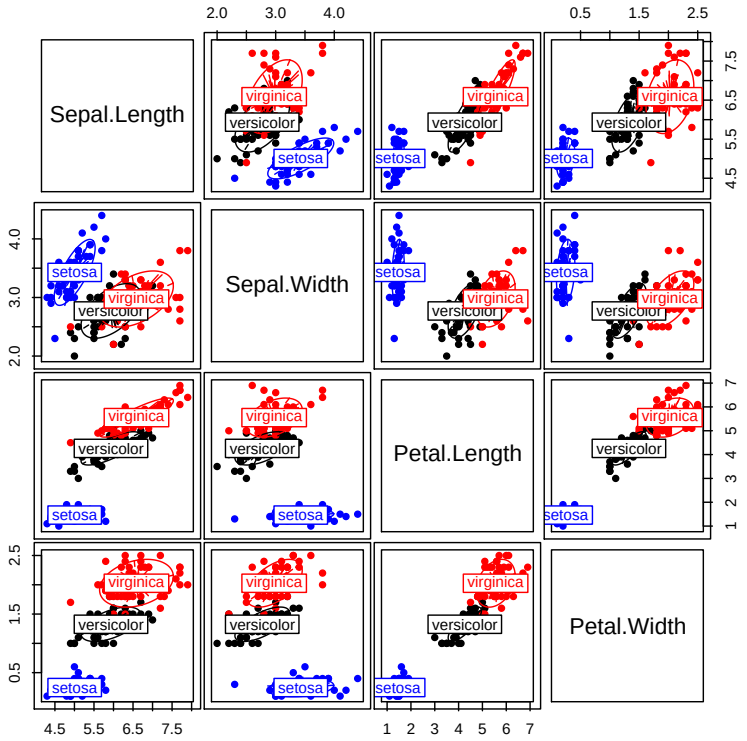
[1] http://cs-people.bu.edu/mdassaro/pp3/




```
par(mfcol = c(3, 4))
for (k in 1:4) {
    j0 <- names(iris)[k]
    br0 <- seq(min(iris[, k]), max(iris[, k]), le = 11)
    x0 <- seq(min(iris[, k]), max(iris[, k]), le = 50)
    for (i in 1:3) {
        i0 <- levels(iris$Species)[i]
        x <- iris[iris$Species == i0, j0]
        hist(x, br = br0, proba = T, col = grey(0.8), main = i0,
            xlab = j0)
        lines(x0, dnorm(x0, mean(x), sd(x)), col = "red", lwd = 2)
    }
}
```

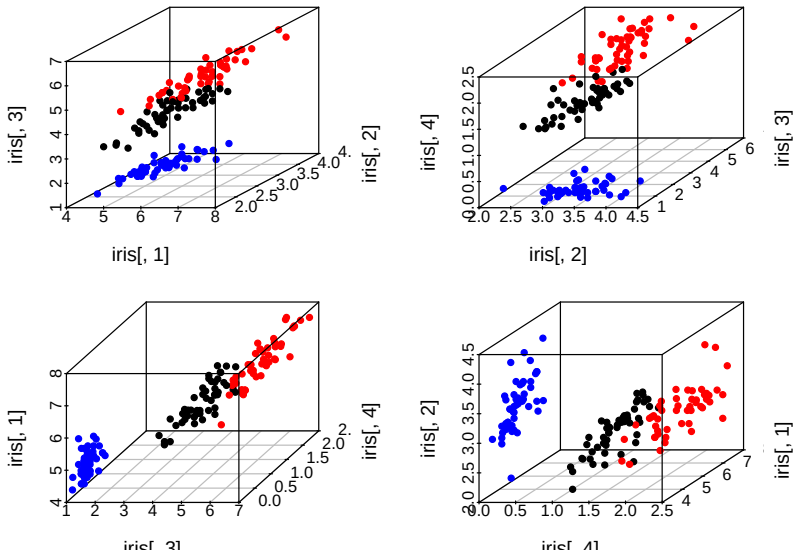
One can display the bivariate scatterplots.

```
library(ade4)
par(mar = c(0, 0, 0, 0))
pan1 <- function(x, y, ...) {
    xy <- cbind.data.frame(x, y)
    s.class(xy, iris$Species, include.ori = F, add.p = T, clab = 1.5,
        col = c("blue", "black", "red"), cpoi = 2, csta = 0.5)
}
pairs(iris[, 1:4], panel = pan1)
```



One can display the 3-dimensional scatterplots.

```
library(scatterplot3d)
par(mfrow = c(2, 2))
mar0 = c(2, 3, 2, 3)
scatterplot3d(iris[, 1], iris[, 2], iris[, 3], mar = mar0, color = c("blue",
    "black", "red")[iris$Species], pch = 19)
scatterplot3d(iris[, 2], iris[, 3], iris[, 4], mar = mar0, color = c("blue",
    "black", "red")[iris$Species], pch = 19)
scatterplot3d(iris[, 3], iris[, 4], iris[, 1], mar = mar0, color = c("blue",
    "black", "red")[iris$Species], pch = 19)
scatterplot3d(iris[, 4], iris[, 1], iris[, 2], mar = mar0, color = c("blue",
    "black", "red")[iris$Species], pch = 19)
```



The ability to discriminate species using this graph approach varies with the ID of the variables and the number of variables considered. Moreover, we can't plot more than 3 variables at the same time.

Looking for a variable or a combination of variables which separate the groups is discriminating. Two main objectives can be distinguished:
1. a descriptive discrimination to answer the question: Which variables separate the groups ?
2. a predictive discrimination to solve the following problem: Let's say I sample a new individual (e.g., a new flower). Which group does it belong to ?

These two aims lead to different ways of programming the discriminant analysis: 1. `discrimin` from the `ade4` package; 2. `lda` from the `MASS` package.

2 The principle

2.1 Linking one variable and a factor

To study a group effect (the factor), one can compute a one-way analysis of variance. We previously saw (cf Within and Between PCA) that such procedure allows:

- ⋆ to answer the null hypothesis: the g population means are equal
- ⋆ to decompose the total sum of square distances to the general mean into between and within sums of squares: $c = b + w$

Let's divide the previous formula by n (or n-1). One can say that the **total variance** c/n can be divided into:

- the variance linked to the factor called **between variance** b/n,
- the variance not linked to the factor called residual variance or **within variance** w/n.

1. From a descriptive point of view, one can compute the correlation ratio: b/c. The correlation ratio varies between 0 and 1. If near 1, the continuous variable and the groups are linked, i.e. the means differ. If near 0, there is no link between the continous variable considered and the groups, i.e. all means are equal.

2. From a predictive point of view, a test (analysis of variance) can be computed based on the weighting ratio b/w.

2.2 Linking a set of p variables and a factor

Let's call $\mathbf{X}$ the data frame containing a set of p variables measured on n individuals. We saw that principal component analyses seek linear combinations of variables maximizing the total inertia, i.e. the total variance. The solution is obtained by the diagonalization of the covariance matrix $\mathbf{X}_0^T\mathbf{D}\mathbf{X}_0\mathbf{Q}$ where $\mathbf{X}_0$ is the $\mathbf{X}$ matrix centred, $\mathbf{D}$ and $\mathbf{Q}$ the weightings of rows and columns respectively.

Let's call $\mathbf{C}$ the **total covariance matrix**. $\mathbf{C}$ can be expressed as a function of:
- a covariance matrix of the table containing the means per group, also called **between covariance matrix B**
- a covariance matrix of the table containing the distances between the individuals and their group means, also called **within covariance matrix W**.

The equation of the one-way analysis of variance can be extended to the total covariance matrix:

$$\mathbf{C} = \mathbf{B} + \mathbf{W}$$

A discriminant analysis looks for combinations of variables $\mathbf{y}$ maximizing the between covariance matrix divided by the total covariance matrix (or divided by the within covariance matrix) under one condition which depends of the approach.
1. If the approach is descriptive: the between covariance matrix is divided by the total covariance matrix $\mathbf{BC}^{-1}$ and the constraint is that the total variance of $\mathbf{y}$ equals 1.
2. If the approach is predictive: the between covariance matrix is divided by the within covariance matrix $\mathbf{BW}^{-1}$ and the constraint is that the within variance of $\mathbf{y}$ equals 1.

The two processes give the same discriminant functions (nearly a constant) and the eigenvalues are linked by a simple relation.

Let's call λ_k an eigenvalue of $\mathbf{BC}^{-1}$ and μ_k the corresponding eigenvalue of $\mathbf{BW}^{-1}$.

$$\mu_k = \frac{\lambda_k}{1-\lambda_k} \Leftrightarrow \lambda_k = \frac{\mu_k}{1+\mu_k}$$

3 Illustration with the Iris dataset

In the Iris dataset, `iris[,1:4]` contains the measures on petals and sepals and `iris$Species` is the categorical variable with the three species: *Iris setosa*, *versicolor*, and *virginica*.

3.1 Using `discrimin` of the `ade4` package

This process is the descriptive approach of the linear discriminant analysis (DA). To compute a DA in the `ade4` package, one uses the `discrimin` function. Before computing a DA, a classical principal component analysis (`dudi.pca`) is performed on the continuous variables to get the table of normed variables, the weightings of rows and columns. Then, the chosen categorical variable is defined in the `discrimin` function.

```
library(ade4)
pca1 <- dudi.pca(iris[, 1:4], scannf = FALSE)
dis1 <- discrimin(pca1, iris$Species, scannf = FALSE)
names(dis1)
```

```
[1] "eig"  "nf"   "fa"   "li"   "va"   "cp"   "gc"   "call"
```

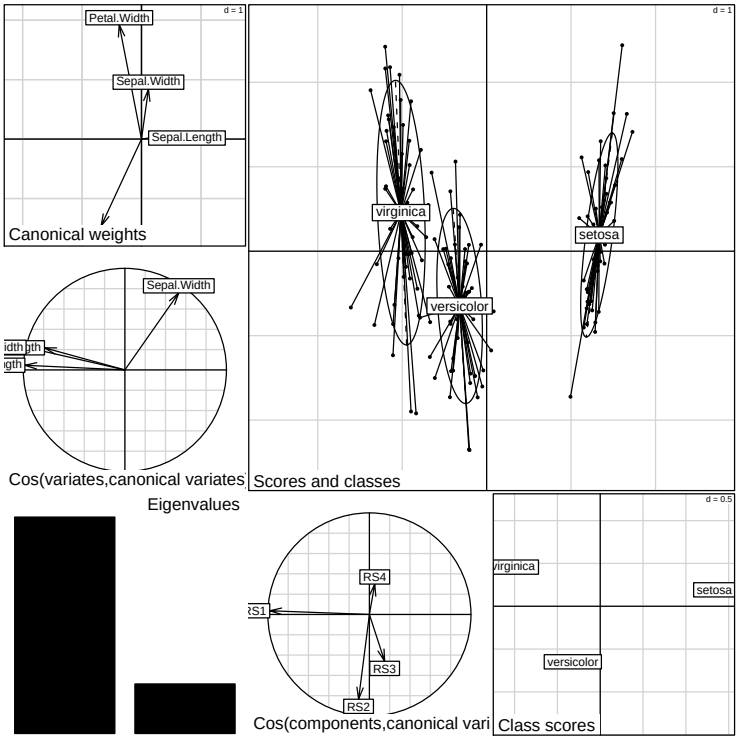
```
dis1
```

```
Discriminant analysis
call: discrimin(dudi = pca1, fac = iris$Species, scannf = FALSE)
class: discrimin
$nf (axis saved) : 2

eigen values: 0.9699 0.222

  data.frame nrow ncol content
1 $fa        4    2    loadings / canonical weights
2 $li        150  2    canonical scores
3 $va        4    2    cos(variables, canonical scores)
4 $cp        4    2    cos(components, canonical scores)
5 $gc        3    2    class scores
```

```
plot(dis1)
```



Six plots are displayed.

1. `Canonical weights` The first scatterplot (top left) represents the coefficients of the linear discriminant functions on the two first axes of the DA. Their total variances equal 1 and their between variances are maximal. The 4 used variables are the normed columns of the PCA.

2. `cos(variates, canonical variates)` The second scatterplot (just below the first one) represents the covariances between the 4 variables and the two first axes of the DA.

3. `Eigenvalues` is the screeplot of the eigenvalues describing the contribution of each axis to the inertia.

4. `Scores and Classes` This plot shows the projections of the individuals onto the plane defined by the axes of the DA. Groups are displayed by ellipses where the centres are the means (between variances) and the ellipses the within variances.

5. `cos(components, canonica variates)` shows the projection of the four axes kept by the normed PCA onto the two axes from the DA.

6. `Class scores` This plot shows the position of the group means on the two first axes of the DA.

3.2 Using `lda` of the `MASS` package

We are here interested in the predictive approach associated to the linear discriminant analysis (DA). To compute a DA in the `MASS` package, one uses the

`lda` function. There is only one step here (as opposed to the two steps procedure seen previously (dudi.pca/discrimin)) and information on the continuous variables (in a object matrix) and the categorical variable considered need to be simultaneously provided.

```
library(MASS)
dis2 <- lda(as.matrix(iris[, 1:4]), iris$Species)
names(dis2)
```

```
[1] "prior"   "counts"  "means"   "scaling" "lev"     "svd"     "N"       "call"
```

```
dis2
```

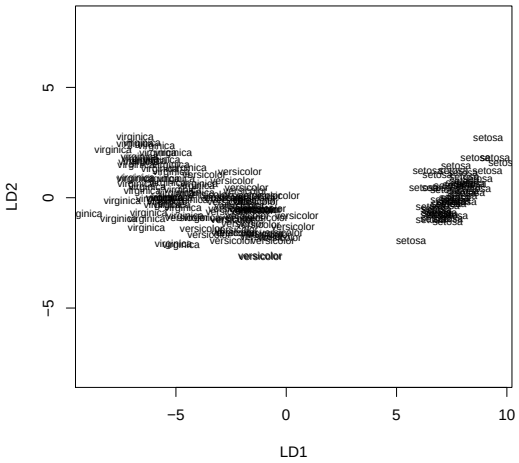
```
Call:
lda(as.matrix(iris[, 1:4]), grouping = iris$Species)
Prior probabilities of groups:
    setosa versicolor  virginica
 0.3333333  0.3333333  0.3333333

Group means:
           Sepal.Length Sepal.Width Petal.Length Petal.Width
setosa            5.006       3.428        1.462       0.246
versicolor        5.936       2.770        4.260       1.326
virginica         6.588       2.974        5.552       2.026

Coefficients of linear discriminants:
                    LD1         LD2
Sepal.Length  0.8293776  0.02410215
Sepal.Width   1.5344731  2.16452123
Petal.Length -2.2012117 -0.93192121
Petal.Width  -2.8104603  2.83918785

Proportion of trace:
   LD1    LD2
0.9912 0.0088
```

```
plot(dis2)
```



Only one scatter plot is displayed: the individual positions on the two discriminant axes.

3.3 Comparison between the two functions

We previously said that the linear functions are defined by a nearly constant value.
`discrimin` provides a linear combination of the normed variables with the `fa` coefficients. The matricial product is provided in `w1`, and if we have a look at the first 10 individuals and the last 10 individuals, we get:

```
w1 <- as.vector(scalewt(iris[, 1:4]) %*% dis1$fa[, 1])
round(w1[1:10], dig = 4)
```

```
 [1] 1.4135 1.2499 1.3132 1.1946 1.4259 1.3504 1.2646 1.3335 1.1503 1.2875
```

```
round(w1[140:150], dig = 4)
```

```
 [1] -0.9124 -1.1665 -0.8952 -0.9657 -1.1916 -1.2006 -0.9898 -0.9082 -0.8710 -1.0321
[11] -0.8211
```

`lda` provides a linear combination of the initial variables with the `scaling` coefficients. The matricial product is provided in `w2`, and if we have a look at the first 10 individuals and the last 10 individuals, we get:

```
w2 <- as.vector(as.matrix(iris[, 1:4]) %*% dis2$scaling[, 1])
round(w2[1:10], dig = 4)
```

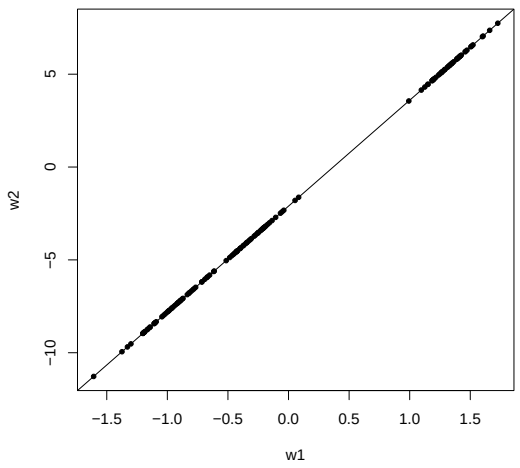
```
 [1] 5.9567 5.0236 5.3847 4.7081 6.0272 5.5968 5.1075 5.5002 4.4554 5.2380
```

```
round(w2[140:150], dig = 4)
```

```
 [1] -7.3089 -8.7582 -7.2107 -7.6126 -8.9011 -8.9525 -7.7501 -7.2847 -7.0728 -7.9913
[11] -6.7883
```

The comparison leads to the following proportional link:

```
plot(w1, w2, pch = 20)
abline(lm(w2 ~ w1))
```



`discrimin` gives a linear combination of total variance equals to 1,

```
var(w1) * 149/150
```

```
[1] 1
```

maximizing the between variance (equal to the first eigenvalue).

```
summary(lm(w1 ~ iris[, 5]))$r.squared
```
```
[1] 0.9698722
```
```
dis1$eig
```
```
[1] 0.9698722 0.2220266
```

`lda` gives a linear combination of within variance equals to 1,

```
tapply(w2, iris[, 5], var)
```
```
    setosa versicolor  virginica
0.7181898  1.0736485  1.2081617
```
```
mean(tapply(w2, iris[, 5], var))
```
```
[1] 1
```

maximizing the 'same' between variance.

```
summary(lm(w2 ~ iris[, 5]))$r.squared
```
```
[1] 0.9698722
```

Eigenvalues of both analyses are linked by the relation $\mu_k = \frac{\lambda_k}{1-\lambda_k}$

```
eigval1 <- dis1$eig
eigval1
```
```
[1] 0.9698722 0.2220266
```
```
eigval2 <- eigval1/(1 - eigval1)
eigval2
```
```
[1] 32.1919292  0.2853910
```
```
eigval2/sum(eigval2)
```
```
[1] 0.991212605 0.008787395
```
```
dis2$svd^2/sum(dis2$svd^2)
```
```
[1] 0.991212605 0.008787395
```

4 More information: tests and allocation

4.1 Testing the eigenvalues

Both procedures (discrimin and lda) allow to test the existence of a true difference between groups, based on randomizations. The null hypothesis is that each group is a random sample of a multinormal distribution.
If the discriminant value (i.e., the eigenvalue) associated to the discriminant function is large enough, the null hypothesis is rejected.

In other words, the generalisation of the one-way ANalysis Of VAriance (ANOVA) is the one-way Multivariate ANalysis Of VAriance (MANOVA). The comparison of the group means for one variable becomes the comparison of group means for a vector of variables.

$$\underset{\text{ANOVA}}{\mu_1 = \mu_2 = ... = \mu_g} \qquad \underset{\text{MANOVA}}{\begin{pmatrix} \mu_1^1 \\ \mu_1^2 \\ \vdots \\ \mu_1^p \end{pmatrix} = \begin{pmatrix} \mu_2^1 \\ \mu_2^2 \\ \vdots \\ \mu_2^p \end{pmatrix} = ... = \begin{pmatrix} \mu_g^1 \\ \mu_g^2 \\ \vdots \\ \mu_g^p \end{pmatrix}}$$

where μ_k is the mean of the variable for the k population and μ_k^j is the mean of the j variable for the k population.

```
measures <- as.matrix(iris[, 1:4])
resm <- manova(measures ~ iris$Species)
```

There are several ways to test the significativity of the eigenvalues (all based on the MANOVA approach):

1. Pillai's test.

```
summary(resm, test = "Pillai")
```

```
             Df Pillai approx F num Df den Df    Pr(>F)
iris$Species   2  1.192   53.466      8    290 < 2.2e-16 ***
Residuals    147
---
Signif. codes:  0
```

The Pillai criteria is the sum of the eigenvalues provided by the `dicrimin` function.

```
sum(eigval1)
```

```
[1] 1.191899
```

2. Wilks's test.

```
summary(resm, test = "Wilks")
```

```
             Df   Wilks approx F num Df den Df    Pr(>F)
iris$Species   2   0.023  199.145      8    288 < 2.2e-16 ***
Residuals    147
---
Signif. codes:  0
```

The Wilks criteria is the product of the within variances provided by the `dicrimin` function.

```
prod(1 - eigval1)
```

```
[1] 0.02343863
```

3. Hotelling-Lawley's test.

```
summary(resm, test = "Hotelling-Lawley")
```

```
             Df Hotelling-Lawley approx F num Df den Df    Pr(>F)
iris$Species   2            32.48   580.53      8    286 < 2.2e-16 ***
Residuals    147
---
Signif. codes:  0
```

The Hotelling-Lawley criteria is the sum of the eigenvalues provided by the `lda` function.

```
sum(eigval2)
```

```
[1] 32.47732
```

4. Roy's test.

```
summary(resm, test = "Roy")
```

```
             Df     Roy approx F num Df den Df    Pr(>F)
iris$Species  2   32.19  1166.96      4    145 < 2.2e-16 ***
Residuals   147
---
Signif. codes:  0
```

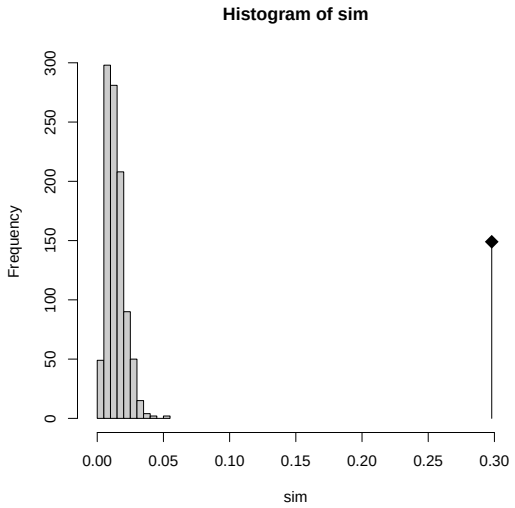
The Roy criteria is the greater eigenvalue provided by the `lda` function.

```
max(eigval2)
```

```
[1] 32.19193
```

If the multinormality is not acceptable (and the challenge is to define what's "acceptable" or not), one can compute a non parametric version of Pillai's test.

```
plot(randtest.discrimin(dis1))
```



4.2 Individual allocations

The `lda` functions allows to answer the following question: knowing the measures of a new individual, can we predict the group it belongs to ?

To understand the process, one can take an example using the Iris data.

⋆ We extract by randomization 50 individual iris from the dataset, create a table containing these individuals `tabref` and a factor containing the name of the species they belong to `espref`.

```
echa <- sample(1:150, 50)
tabref <- iris[echa, 1:4]
espref <- iris[echa, 5]
```

⋆ We create a table `tabsup` with the 100 other iris and a factor containing the names of the species they belong to `espsup`.

```
tabsup <- iris[-echa, 1:4]
espsup <- iris[-echa, 5]
```

⋆ We compute a linear discriminant analysis using the 50 references (using tabref espref).

```
lda0 <- lda(tabref, espref)
lda0

Call:
lda(tabref, espref)
Prior probabilities of groups:
    setosa versicolor  virginica
      0.40       0.26       0.34

Group means:
           Sepal.Length Sepal.Width Petal.Length Petal.Width
setosa         5.010000    3.380000     1.470000    0.255000
versicolor     5.969231    2.846154     4.323077    1.369231
virginica      6.670588    2.988235     5.700000    2.029412

Coefficients of linear discriminants:
                    LD1         LD2
Sepal.Length  0.1124365  0.0746533
Sepal.Width   2.3258368  2.6838519
Petal.Length -1.7955507 -0.3309531
Petal.Width  -3.5351688  1.4259364

Proportion of trace:
   LD1    LD2
0.9968 0.0032
```

⋆ We predict the allocations of the 100 other iris and provide a contingency table dealing with the known species information and the allocations.

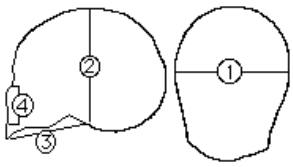
```
espestim <- predict(lda0, tabsup)$class
table(espestim, espsup)

            espsup
espestim     setosa versicolor virginica
  setosa         30          0         0
  versicolor      0         36         2
  virginica       0          1        31
```

We can predict the allocation, analyse the predictions (very good with this dataset) but there is no specific test to calculate an error of misclassification.

5 Your turn!

The data were proposed by Manly [3].
The `skulls` dataframe has 150 rows (egyptian skulls) and 4 columns (anthropometric measures). The four variables are the maximum breadth (V1), the basibregmatic height (V2), the basialveolar length (V3) and the nasal height (V4). All measurements are expressed in millimeters.



The measurements are made on 5 groups (30 skulls per group). The groups are defined as follows :
1 - the early predynastic period (circa 4000 BC)
2 - the late predynastic period (circa 3300 BC)
3 - the 12th and 13th dynasties (circa 1850 BC)
4 - the Ptolemiac period (circa 200 BC)
5 - the Roman period (circa 150 BC).
The group vector is obtained using the `gl` function (look at the help to better understand this useful function) and the `levels` function to associate a name to each modality.

```
fac <- gl(5, 30)
levels(fac) <- c("-4000", "-3300", "-1850", "-200", "+150")
```

6 Conclusion

The linear discriminant analysis is well-known and well-described especially due to the old statistical debate between exploratory and confirmatory methods. Its use depends on the situation, data and objectives. Always remember that statistical solutions are defined by ecological questions - meaningless results are often linked to badly formulated questions.
R provides several methods to deal with the discriminant analysis such as `discrimin` associated to the exploratory point of view and `lda` to the confirmatory point of view.

method	**discrimin**	**lda**
linear combination of variables	fa	LD
variance of the linear combination	total variance equals 1	within variance equals 1
maximized criteria	$\mathbf{BC}^{-1}$	$\mathbf{BW}^{-1}$

References

[1] E. Anderson. The irises of the gaspe peninsula. *Bulletin of the American Iris Society*, 59:2–5, 1935.

[2] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[3] B.F.J. Manly. *Randomization and Monte Carlo methods in biology*. Chapman and Hall, London, 1991.