

Multivariate Statistical Analysis - Exercise Session 9

10.03.2022

Problem 1: Linear Discriminant Analysis

Install package MASS if you haven't yet.

```
install.packages("MASS")
```

Import package MASS and read the data. Leave only species versicolor and virginica, i.e., drop first 50 observations. We also drop redundant level setosa with the function `droplevels`.

```
library(MASS)
data(iris)
```

```
iris <- droplevels(iris[-(1:50), ])
dim(iris)
```

```
## [1] 100  5
```

```
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width  Species
## 51          7.0         3.2         4.7         1.4 versicolor
## 52          6.4         3.2         4.5         1.5 versicolor
## 53          6.9         3.1         4.9         1.5 versicolor
## 54          5.5         2.3         4.0         1.3 versicolor
## 55          6.5         2.8         4.6         1.5 versicolor
## 56          5.7         2.8         4.5         1.3 versicolor
```

Firstly, we visualize the data in Figure 1.

```
pairs(iris[, 1:4], pch = c(16, 17)[iris$Species], gap = 0, upper.panel = NULL,
      col = c(rgb(0, 0, 1, 0.5), rgb(1, 0, 0, 0.5))[iris$Species])
par(xpd = TRUE)
legend(0.75, 0.75, legend = levels(iris$Species), pch = c(16, 17),
      col = c(rgb(0, 0, 1, 0.5), rgb(1, 0, 0, 0.5)), cex = 1)
```

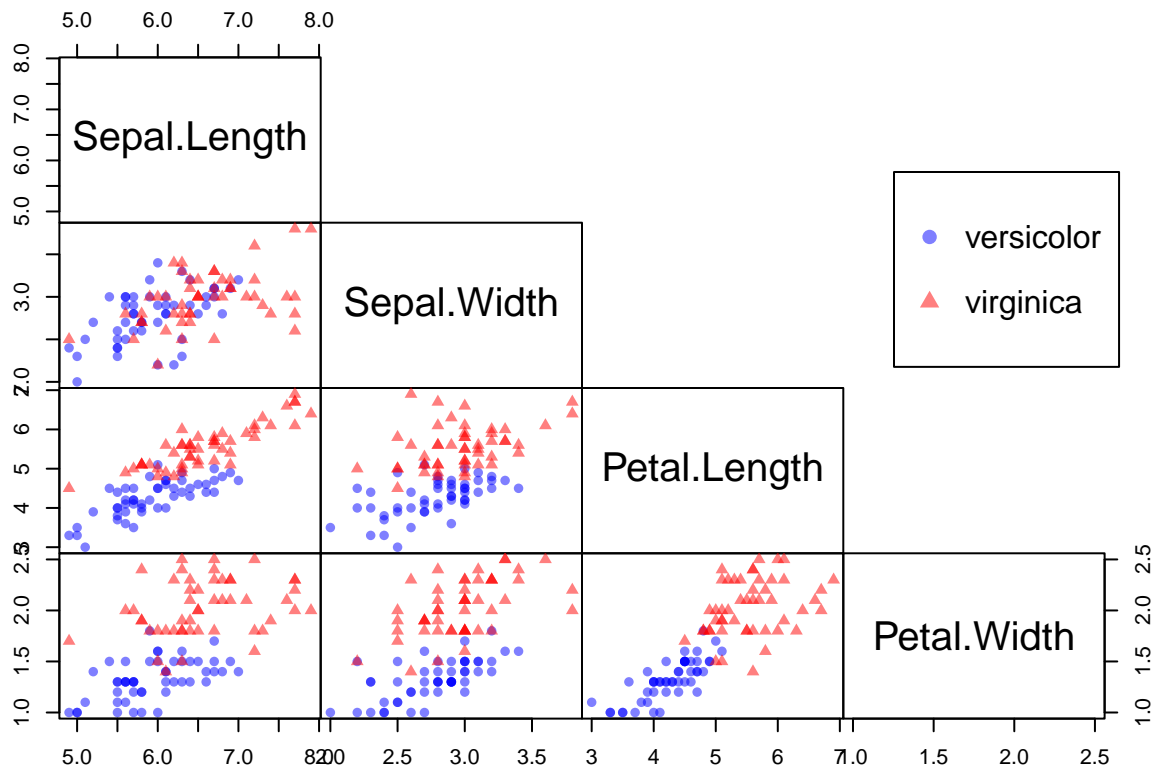


Figure 1: Pairwise scatter plots of variables.

a) Perform Fisher's linear discriminant analysis

With function `lda` from package `MASS` we can perform Fisher's linear discriminant analysis. Function `lda` can be used quite similarly compared to the function `lm`. That is, the input can be given as a formula

$$y \sim x_1 + x_2 + \dots + x_g,$$

where response y is the grouping factor and x_1, x_2, \dots, x_g are the discriminators. In this case,

```

y = Species,
x1 = Sepal.Length,
x2 = Sepal.Width,
x3 = Petal.Length and
x4 = Petal.Width.

```

Below we perform Fisher's linear discriminant analysis and give the vector a .

```

# Species ~ . is a shorthand notation for
# Species ~ Sepal.Length + Sepal.Width + Petal.Length + Petal.Width
iris_lda <- lda(Species ~ ., data = iris)
a_lda <- iris_lda$scaling
a_lda

```

```
##                LD1
## Sepal.Length -0.9431178
## Sepal.Width  -1.4794287
## Petal.Length  1.8484510
## Petal.Width   3.2847304
```

Next we perform Fisher's linear discriminant analysis manually. Let

$$W = \sum_{i=1}^g (n_i - 1) \text{Cov}(X_i) \quad \text{and}$$

$$B = \sum_{i=1}^g n_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T.$$

In the case when $g = 2$, we can express B as

$$B = \frac{n_1 n_2}{n} d d^T,$$

where $d = \bar{x}_1 - \bar{x}_2$. Then vector a is obtained by computing

$$a = W^{-1}d.$$

In general case $g \in \{2, 3, \dots\}$, solution a is equal to the eigenvector corresponding to the largest eigenvalue of $W^{-1}B$.

```
# cov(X_1)
s1 <- cov(iris[1:50, 1:4])

# cov(X_2)
s2 <- cov(iris[51:100, 1:4])

d1 <- colMeans(iris[1:50, 1:4])
d2 <- colMeans(iris[51:100, 1:4])

d <- d1 - d2
d <- as.matrix(d, ncol = 1)

b <- (50 * 50) / 100 * d %*% t(d)
w <- 49 * (s1 + s2)
l <- solve(w) %*% b

# Eigenvectors of l
a_manual <- as.numeric(eigen(l)$vectors[, 1])
a_manual2 <- solve(w) %*% d
```

By comparing `a_lda`, `a_manual` and `a_manual2` we can see that they are equal up to scale.

```
c(norm(a_lda, type = "2"), norm(a_manual, type = "2"),
  norm(a_manual2, type = "2"))

## [1] 4.1574518 1.0000000 0.1599683

a_lda <- a_lda / norm(a_lda, type = "2")
a_manual2 <- a_manual2 / norm(a_manual2, type = "2")

data.frame(a_lda = a_lda, a_manual = a_manual, a_manual2 = a_manual2)
```

```
##           LD1   a_manual a_manual2
## Sepal.Length -0.2268500 -0.2268500  0.2268500
## Sepal.Width  -0.3558499 -0.3558499  0.3558499
## Petal.Length  0.4446115  0.4446115 -0.4446115
## Petal.Width   0.7900826  0.7900826 -0.7900826
```

b) In which group will new flower be classified?

```
newobs <- data.frame(Sepal.Length = 6, Sepal.Width = 3, Petal.Length = 4,
                    Petal.Width = 1)
predict(iris_lda, newdata = newobs)$class
```

```
## [1] versicolor
## Levels: versicolor virginica
```

New flower will be classified as `versicolor`.

We can also predict the group of new observation x by hand. New observation x is allocated to the population whose mean score is closest to the $a^T x$. That is, x is allocated to group j , if

$$|a^T x - a^T \bar{x}_j| < |a^T x - a^T \bar{x}_i|, \quad \text{for all } i \neq j.$$

```
# new observation is classified as versicolor
abs(t(a_lda) %*% (as.numeric(newobs) - d1)) <
  abs(t(a_lda) %*% (as.numeric(newobs) - d2))
```

```
##      [,1]
## LD1 TRUE
```

c) Leave-one-out cross-validation

Below we perform leave-one-out cross-validation

```
d_cv <- lda(Species ~ ., data = iris, CV = TRUE)
```

Results of leave-one-out cross-validation can be cross tabulated. Here two `versicolor` are classified as `virginica` and one `virginica` is classified as `versicolor`.

```
result <- data.frame(est = d_cv$class, truth = iris[, 5])
table(result)
```

```
##           truth
## est      versicolor virginica
## versicolor      48          1
## virginica       2          49
```

Thus misclassification rate is $3/100 = 0.03$. We can also perform leave-one-out cross-validation manually.

```
predicted <- rep(NA, 100)
for (i in 1:100) {
  train <- iris[-i, ]
  test  <- iris[i, ]
  predicted[i] <- predict(lda(Species ~ ., data = train), newdata = test)$class
}
predicted <- factor(predicted, levels = c(1, 2),
                  labels = c("versicolor", "virginica"))
sum(predicted != iris$Species) / nrow(iris)
```

```
## [1] 0.03
```

Problem 2: Fisher's Linear Discriminant Function

Show that the solution for the problem:

$$\max_a \left\{ \frac{a^T B a}{a^T W a} \right\},$$

is obtained by setting a equal to the eigenvector of $W^{-1}B$ that corresponds to the largest eigenvalue.

W = measure of group dispersions B = dispersion between groups

Since the vector a can be scaled arbitrarily without affecting the ratio, we can formulate the problem as follows:

$$\max_a \{a^T B a\} \quad \text{s.t.} \quad a^T W a = 1.$$

Let $W^{1/2}$ be the symmetric square root of W . Note that matrix W is symmetric. Let $z = W^{1/2}a$ and $a = W^{-1/2}z$. Then

$$\begin{aligned} a^T B a &= (W^{-1/2}z)^T B (W^{-1/2}z) = z^T W^{-1/2} B W^{-1/2} z \quad \text{and} \\ a^T W a &= (W^{-1/2}z)^T W (W^{-1/2}z) = z^T \underbrace{W^{-1/2} W W^{-1/2}}_{=I} z = z^T z. \end{aligned}$$

Note that $W^{-1/2} B W^{-1/2}$ is symmetric and hereby the spectral decomposition exists ($W^{-1/2} B W^{-1/2} = \Gamma \Lambda \Gamma^T$ and $\Gamma^T \Gamma = I$). Denote $w = \Gamma^T z$. Then

$$\begin{aligned} z^T W^{-1/2} B W^{-1/2} z &= z^T \Gamma \Lambda \Gamma^T z = w^T \Lambda w, \quad \text{and} \\ z^T z &= z^T \Gamma \Gamma^T z = w^T w. \end{aligned}$$

Now we can reformulate the problem as

$$\max_w \{w^T \Lambda w\} = \max_w \left\{ \sum_{i=1}^p \lambda_i w_i^2 \right\} \quad \text{s.t.} \quad w^T w = 1.$$

Since $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p$, we choose the first element of w to be one and the rest to be zero. This means that $z = \Gamma w = \gamma_1$, where γ_1 is the first eigenvector of $W^{-1/2} B W^{-1/2}$ and $a = W^{-1/2}z = W^{-1/2}\gamma_1$.

Note that for any two matrices $A \in \mathbb{R}^{n \times p}$ and $C \in \mathbb{R}^{p \times n}$, the non-zero eigenvalues of AC and CA are the same and have the same multiplicity (Theorem A.6.2 of Mardia, Kent and Bibby). Now let $A = W^{-1/2}B$ and $C = W^{-1/2}$, this means that the non-zero eigenvalues of $CA = W^{-1}B$ are the same as $AC = W^{-1/2} B W^{-1/2}$. Hence, λ_1 is the largest eigenvalue of $W^{-1}B$. Since γ_1 is the eigenvector corresponding to the largest λ_1 of $W^{-1/2} B W^{-1/2}$, we have that

$$W^{-1}B(W^{-1/2}\gamma_1) = W^{-1/2}(W^{-1/2} B W^{-1/2}\gamma_1) = W^{-1/2}\lambda_1\gamma_1 = \lambda_1(W^{-1/2}\gamma_1).$$

This shows that $a = W^{-1/2}\gamma_1$ is the eigenvector of $W^{-1}B$ corresponding to its largest eigenvalue λ_1 .