Estadística descriptiva multivariante
Soluciones

Francesc Carmona y Josep Gregori*

28 de febrero de 2018

1. Estadísticos

Ejercicio 1

*Para la base de datos crabs del paquete MASS de **R**:*

a) *Estudiar la base de datos con la ayuda y las funciones `str()` y `summary()`.*

```
> library(MASS)
> data(crabs)
> help(crabs)
> str(crabs)

'data.frame': 200 obs. of  8 variables:
 $ sp   : Factor w/ 2 levels "B","O": 1 1 1 1 1 1 1 1 1 1 ...
 $ sex  : Factor w/ 2 levels "F","M": 2 2 2 2 2 2 2 2 2 2 ...
 $ index: int  1 2 3 4 5 6 7 8 9 10 ...
 $ FL   : num  8.1 8.8 9.2 9.6 9.8 10.8 11.1 11.6 11.8 11.8 ...
 $ RW   : num  6.7 7.7 7.8 7.9 8 9 9.9 9.1 9.6 10.5 ...
 $ CL   : num  16.1 18.1 19 20.1 20.3 23 23.8 24.5 24.2 25.2 ...
 $ CW   : num  19 20.8 22.4 23.1 23 26.5 27.1 28.4 27.8 29.3 ...
 $ BD   : num  7 7.4 7.7 8.2 8.2 9.8 9.8 10.4 9.7 10.3 ...

> summary(crabs)

 sp      sex         index            FL              RW       
 B:100   F:100   Min.   : 1.0   Min.   : 7.20   Min.   : 6.50  
 O:100   M:100   1st Qu.:13.0   1st Qu.:12.90   1st Qu.:11.00  
                 Median :25.5   Median :15.55   Median :12.80  
                 Mean   :25.5   Mean   :15.58   Mean   :12.74  
                 3rd Qu.:38.0   3rd Qu.:18.05   3rd Qu.:14.30  
                 Max.   :50.0   Max.   :23.10   Max.   :20.20  
       CL              CW              BD       
 Min.   :14.70   Min.   :17.10   Min.   : 6.10  
 1st Qu.:27.27   1st Qu.:31.50   1st Qu.:11.40  
 Median :32.10   Median :36.80   Median :13.90  
 Mean   :32.11   Mean   :36.41   Mean   :14.03  
 3rd Qu.:37.23   3rd Qu.:42.00   3rd Qu.:16.60  
 Max.   :47.60   Max.   :54.60   Max.   :21.60  
```

*Alumno del curso 2009-10

```
> attach(crabs)
> crabs5 <- crabs[,4:8]  # las 5 variables numéricas
```

b) *Calcular estadísticos descriptivos como la media, la mediana, la varianza, etc. de las variables numéricas, tanto para todos los datos, como para las especies y los sexos por separado.*

```
> # Medias por especie de cada variable numérica
> apply(crabs5, MARGIN = 2, FUN = tapply, INDEX=sp, mean)

      FL     RW     CL     CW     BD
B 14.056 11.928 30.058 34.717 12.583
O 17.110 13.549 34.153 38.112 15.478

> # Medias por sexo de cada variable numérica
> apply(crabs5, MARGIN = 2, FUN = tapply, INDEX=sex, mean)

      FL     RW     CL     CW     BD
F 15.432 13.487 31.360 35.830 13.724
M 15.734 11.990 32.851 36.999 14.337

> # Medianas por especie de cada variable numérica
> apply(crabs5, MARGIN = 2, FUN = tapply, INDEX=sp, median)

     FL    RW    CL    CW    BD
B 14.45 12.00 30.10 35.20 12.60
O 17.50 13.65 34.55 38.95 15.55

> # Medianas por sexo de cada variable numérica
> apply(crabs5, MARGIN = 2, FUN = tapply, INDEX=sex, median)

     FL    RW    CL    CW   BD
F 15.45 13.80 31.70 36.45 13.8
M 15.70 11.85 32.45 37.10 14.2

> # Desviaciones típicas por especie de cada variable numérica
> apply(crabs5, MARGIN = 2, FUN = tapply, INDEX=sp, sd)

        FL       RW       CL       CW       BD
B 3.019610 2.279291 6.902703 7.866038 3.067887
O 3.275575 2.605530 6.764262 7.540922 3.151481

> # Desviaciones típicas por sexo de cada variable numérica
> apply(crabs5, MARGIN = 2, FUN = tapply, INDEX=sex, sd)

        FL       RW       CL       CW       BD
F 3.537930 2.740702 6.702992 7.380927 3.343231
M 3.463379 2.160504 7.471204 8.330248 3.494228
```

c) *Hallar los cinco números descriptivos de cada variable numérica. Utilizar la función* `fivenum()`.

```
> cinconum <- apply(crabs5, 2, fivenum)
> row.names(cinconum) <- c("min","lower.h","median","upper.h","max")
> cinconum

            FL   RW    CL   CW   BD
min       7.20  6.5 14.70 17.1  6.1
lower.h  12.90 11.0 27.25 31.5 11.4
median   15.55 12.8 32.10 36.8 13.9
upper.h  18.10 14.3 37.25 42.0 16.6
max      23.10 20.2 47.60 54.6 21.6
```

d) *Hallar las matrices de varianzas y covarianzas de las variables numéricas según especie y sexo.*[1]

```
> round(var(crabs5),2)

      FL    RW    CL    CW    BD
FL 12.22  8.16 24.36 26.55 11.82
RW  8.16  6.62 16.35 18.24  7.84
CL 24.36 16.35 50.68 55.76 23.97
CW 26.55 18.24 55.76 61.97 26.09
BD 11.82  7.84 23.97 26.09 11.73

> # Por sexo F
> round(var(crabs5[sex=="F",]),2)

      FL    RW    CL    CW    BD
FL 12.52  9.38 23.27 25.34 11.65
RW  9.38  7.51 18.12 19.93  8.92
CL 23.27 18.12 44.93 49.30 22.14
CW 25.34 19.93 49.30 54.48 24.14
BD 11.65  8.92 22.14 24.14 11.18

> # Por sexo M
> round(var(crabs5[sex=="M",]),2)

      FL    RW    CL    CW    BD
FL 11.99  7.24 25.46 27.85 12.02
RW  7.24  4.67 15.88 17.62  7.30
CL 25.46 15.88 55.82 61.91 25.58
CW 27.85 17.62 61.91 69.39 27.94
BD 12.02  7.30 25.58 27.94 12.21

> # Por especie B
> round(var(crabs5[sp=="B",]),2)

      FL    RW    CL    CW    BD
FL  9.12  6.18 20.74 23.64  9.16
RW  6.18  5.20 14.10 16.21  6.32
CL 20.74 14.10 47.65 54.22 21.00
CW 23.64 16.21 54.22 61.87 23.95
BD  9.16  6.32 21.00 23.95  9.41
```

[1] Observar que en **R** las funciones matriciales `var()` o `cov()` proporcionan la matriz de varianzas corregida $\widehat{\mathbf{S}} = \frac{n}{n-1}\mathbf{S}$.

```
> # Por especie O
> round(var(crabs5[sp=="O",]),2)

      FL    RW    CL    CW    BD
FL 10.73  7.72 21.90 24.49 10.14
RW  7.72  6.79 15.42 17.67  7.06
CL 21.90 15.42 45.76 50.84 21.19
CW 24.49 17.67 50.84 56.87 23.53
BD 10.14  7.06 21.19 23.53  9.93
```

e) *Comparar cada especie con medidas globales de variabilidad: varianza total* $\mathrm{tr}(\mathbf{S})$, *varianza media* $\mathrm{tr}(\mathbf{S})/p$, *varianza generalizada* $|\mathbf{S}|$ *y varianza efectiva* $|\mathbf{S}|^{1/p}$.

```
> # Varianza total global
> sum(diag(var(crabs5)))

[1] 143.216

> # que equivale a
> sum(eigen(var(crabs5))$values)

[1] 143.216

> # Varianza total para la especie B
> sum(diag(var(crabs5[crabs$sp=="B",])))

[1] 133.247

> # Varianza total para la especie O
> sum(diag(var(crabs5[crabs$sp=="O",])))

[1] 130.0708

> # Varianza media global
> sum(diag(var(crabs5)))/5

[1] 28.64321

> # Varianza media para la especie B
> sum(diag(var(crabs5[crabs$sp=="B",])))/5

[1] 26.6494

> # Varianza media para la especie O
> sum(diag(var(crabs5[crabs$sp=="O",])))/5

[1] 26.01415

> # Varianza generalizada global
> det(var(crabs5))
```

```
[1] 1.924094

> # que equivale a
> prod(eigen(var(crabs5))$values)

[1] 1.924094

> # Varianza generalizada para la especie B
> det(var(crabs5[crabs$sp=="B",]))

[1] 0.07706116

> # Varianza generalizada para la especie O
> det(var(crabs5[crabs$sp=="O",]))

[1] 0.2457478

> # Varianza efectiva global
> det(var(crabs5))^(1/5)

[1] 1.139844

> # que equivale a
> prod(eigen(var(crabs5))$values)^(1/5)

[1] 1.139844

> # Varianza efectiva para la especie B
> det(var(crabs5[crabs$sp=="B",]))^(1/5)

[1] 0.5989176

> # Varianza efectiva para la especie O
> det(var(crabs5[crabs$sp=="O",]))^(1/5)

[1] 0.7552625
```

Ejercicio 2 (∗)

Para la base de datos **huswif** *del libro de Everitt(2005)*[2]

a) *Cargar los datos con la instrucción:*

```
huswif <- source("chap1huswif.dat")$value
```

y estudiar la base de datos con las funciones `str()` *y* `summary()`*.*

b) *Calcular estadísticos descriptivos como la media, la mediana, la varianza, etc. de todas las variables.*

c) *Hallar los cinco números descriptivos de cada variable numérica. Utilizar la función* `fivenum()`*.*

[2] Ver la página web `https://www.york.ac.uk/depts/maths/data/everitt/welcome.htm`.

d) *Hallar la matriz de varianzas-covarianzas y la matriz de correlaciones.*

e) *Hallar las medidas globales de variabilidad: varianza total* $\mathrm{tr}(\mathbf{S})$*, varianza media* $\mathrm{tr}(\mathbf{S})/p$*, varianza generalizada* $|\mathbf{S}|$ *y varianza efectiva* $|\mathbf{S}|^{1/p}$*.*

Es un ejercicio idéntico al anterior.
La matriz de correlaciones se obtiene con la función `cor()`.

Ejercicio 3 (∗)

Para la base de datos ***airpoll*** *del libro de Everitt(2005).*

a) *Cargar los datos con la instrucción:*

```
airpoll <- source("chap2airpoll.dat")$value
```

y estudiar la base de datos con las funciones `str()` *y* `summary()`*.*

b) *Calcular estadísticos descriptivos como la media, la mediana, la varianza, etc. de todas las variables.*

c) *Hallar los cinco números descriptivos de cada variable numérica. Utilizar la función* `fivenum()`*.*

d) *Hallar la matriz de varianzas-covarianzas y la matriz de correlaciones.*

e) *Hallar las medidas globales de variabilidad: varianza total* $\mathrm{tr}(\mathbf{S})$*, varianza media* $\mathrm{tr}(\mathbf{S})/p$*, varianza generalizada* $|\mathbf{S}|$ *y varianza efectiva* $|\mathbf{S}|^{1/p}$*.*

Es un ejercicio idéntico al ejercicio 1.

Ejercicio 4

Entre los 10 elementos muestrales o parejas casadas (con papeles o no) de la base de datos ***huswif*** *del libro de Everitt(2005):*

a) *Calcular las distancias euclídeas con la función* `dist` *de* ***R*** *y expresarlas en forma de matriz.*

```
> huswif <- source("data/chap1huswif.dat")$value
> huswif

   Hage Hheight Wage Wheight Hagefm
1    49    1809   43    1590     25
2    25    1841   28    1560     19
3    40    1659   30    1620     38
4    52    1779   57    1540     26
5    58    1616   52    1420     30
6    32    1695   27    1660     23
7    43    1730   52    1610     33
8    47    1740   43    1580     26
9    31    1685   23    1610     26
10   26    1735   25    1590     23

> attach(huswif)
> # distancia euclídea
> (dd <- round(dist(huswif),2))
```

```
        1      2      3      4      5      6      7      8      9
2   52.55
3  154.33 193.17
4   60.05  76.57 147.71
5  257.56 268.35 206.69 202.60
6  135.81 177.15  56.52 150.88 255.33
7   82.60 126.16  75.23  86.35 222.10  67.61
8   69.76 106.58  92.32  57.81 202.96  94.42  33.85
9  128.46 164.15  32.40 123.83 206.03  51.24  55.31  67.68
10  79.58 110.28  84.39  78.39 211.81  80.87  39.28  29.98  54.20

> as.matrix(dd)

        1      2      3      4      5      6      7      8      9     10
1    0.00  52.55 154.33  60.05 257.56 135.81  82.60  69.76 128.46  79.58
2   52.55   0.00 193.17  76.57 268.35 177.15 126.16 106.58 164.15 110.28
3  154.33 193.17   0.00 147.71 206.69  56.52  75.23  92.32  32.40  84.39
4   60.05  76.57 147.71   0.00 202.60 150.88  86.35  57.81 123.83  78.39
5  257.56 268.35 206.69 202.60   0.00 255.33 222.10 202.96 206.03 211.81
6  135.81 177.15  56.52 150.88 255.33   0.00  67.61  94.42  51.24  80.87
7   82.60 126.16  75.23  86.35 222.10  67.61   0.00  33.85  55.31  39.28
8   69.76 106.58  92.32  57.81 202.96  94.42  33.85   0.00  67.68  29.98
9  128.46 164.15  32.40 123.83 206.03  51.24  55.31  67.68   0.00  54.20
10  79.58 110.28  84.39  78.39 211.81  80.87  39.28  29.98  54.20   0.00
```

b) *Calcular las distancias de K.Pearson y dar la matriz de distancias.*

Según la fórmula 1.3 de la página 19 de Cuadras(2018) la distancia de Pearson es:

$$d_P(i,j) = \sqrt{\sum_{h=1}^{p}(x_{ih} - x_{jh})^2/s_{hh}}$$

donde s_{hh} es la varianza de la variable X_h. Así pues, es como un paso intermedio entre la distancia euclídea, que no tiene en cuenta ni las varianzas ni las covarianzas de las variables, y la distancia de Mahalanobis.

Podemos hacer un cálculo “manual” de esta distancia entre las dos primeras parejas o filas.

```
> vv <- apply(huswif,2,var)
> sqrt(sum((huswif[1,]-huswif[2,])^2/vv))

[1] 2.725315
```

También podemos aprovechar la función `mahalanobis()` y adaptarla para que calcule las distancias de Pearson:

```
> sqrt(mahalanobis(as.vector(huswif[1,], mode="numeric"),
+                  as.vector(huswif[2,], mode="numeric"), cov = diag(vv)))

[1] 2.725315
```

Como nos piden la matriz de distancias dos a dos utilizaremos la función `D2.dist()` del paquete `biotools` que en realidad sirve para calcular la distancia (al cuadrado) de Mahalanobis:

```
> library(biotools)

---
biotools version 3.1

> round(sqrt(D2.dist(huswif, cov = diag(vv))),5)

         1       2       3       4       5       6       7       8       9
2  2.72531
3  3.50720 4.66201
4  1.44330 3.64108 3.86591
5  4.09745 5.60013 4.18887 3.17157
6  2.80049 2.80046 2.95559 3.71319 5.07485
7  2.08138 3.97020 2.22262 2.02435 3.66607 2.98846
8  1.04863 2.99764 2.82833 1.44526 3.37006 2.35527 1.57850
9  2.88322 2.79955 2.43038 3.66849 4.57188 1.01309 2.87806 2.29198
10 2.70681 1.78902 3.26041 3.56663 4.88502 1.34713 3.17716 2.38419 1.06979
```

Hay otro concepto de "distancia" de Pearson basada en la correlación entre variables o series. Sabemos que el coeficiente de correlación entre dos variables aleatorias es una medida de asociación lineal. Lo mismo para dos observaciones (filas) de la muestra. Entonces, la disimilaridad $1 - \mathrm{cor}(\mathbf{x}_i', \mathbf{x}_j')$ mide la diferencia entre esas observaciones y está entre 0 y 2. Esta es la definición que utiliza la función `get_dist()` del paquete `factoextra`.

```
> # Distancias con la correlación de Pearson
> round(1 - cor(t(huswif)), 5)

         1       2       3       4       5       6       7       8       9
1  0.00000 0.00029 0.00245 0.00007 0.00001 0.00256 0.00072 0.00023 0.00156
2  0.00029 0.00000 0.00426 0.00013 0.00029 0.00443 0.00182 0.00099 0.00305
3  0.00245 0.00426 0.00000 0.00329 0.00255 0.00001 0.00057 0.00120 0.00010
4  0.00007 0.00013 0.00329 0.00000 0.00006 0.00338 0.00116 0.00052 0.00223
5  0.00001 0.00029 0.00255 0.00006 0.00000 0.00264 0.00077 0.00025 0.00164
6  0.00256 0.00443 0.00001 0.00338 0.00264 0.00000 0.00060 0.00126 0.00013
7  0.00072 0.00182 0.00057 0.00116 0.00077 0.00060 0.00000 0.00014 0.00020
8  0.00023 0.00099 0.00120 0.00052 0.00025 0.00126 0.00014 0.00000 0.00060
9  0.00156 0.00305 0.00010 0.00223 0.00164 0.00013 0.00020 0.00060 0.00000
10 0.00042 0.00129 0.00087 0.00080 0.00048 0.00094 0.00007 0.00005 0.00037
        10
1  0.00042
2  0.00129
3  0.00087
4  0.00080
5  0.00048
6  0.00094
7  0.00007
8  0.00005
9  0.00037
10 0.00000

> library(factoextra)
> round(get_dist(huswif, method="pearson"),5)
```

```
          1       2       3       4       5       6       7       8       9
2  0.00029
3  0.00245 0.00426
4  0.00007 0.00013 0.00329
5  0.00001 0.00029 0.00255 0.00006
6  0.00256 0.00443 0.00001 0.00338 0.00264
7  0.00072 0.00182 0.00057 0.00116 0.00077 0.00060
8  0.00023 0.00099 0.00120 0.00052 0.00025 0.00126 0.00014
9  0.00156 0.00305 0.00010 0.00223 0.00164 0.00013 0.00020 0.00060
10 0.00042 0.00129 0.00087 0.00080 0.00048 0.00094 0.00007 0.00005 0.00037
```

Otra definición de esta distancia de Pearson basada en la correlación es

$$\delta_P(i,j) = \sqrt{\frac{1 - \operatorname{cor}(\mathbf{x}_i', \mathbf{x}_j')}{2}}$$

Esta es la que utiliza la función `pearson.dist()` del paquete `hyperSpec`.

c) *Calcular la distancia de Mahalanobis (sin cuadrado) entre la primera y la última de las parejas (`huswif[1,]` y `huswif[10,]`).*

```
> # Distancia de Mahalanobis entre la primera y la última pareja.
> sqrt(mahalanobis(as.vector(huswif[1,],  mode="numeric"),
+                  as.vector(huswif[10,], mode="numeric"), cov(huswif)))

[1] 3.409028
```

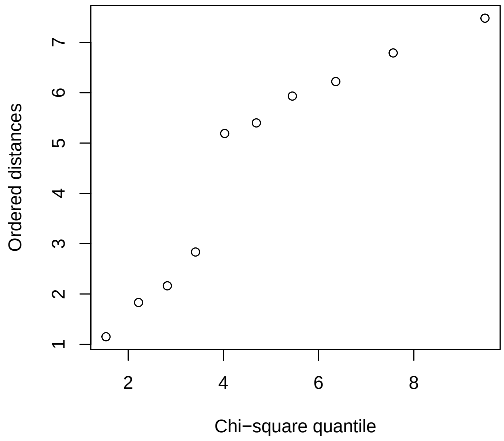
d) *Calcular las distancias al cuadrado de Mahalanobis de cada pareja al vector de medias y su correspondiente matriz de covarianzas. Probar `?mahalanobis` en **R**.*

```
> # Distancias al cuadrado de Mahalanobis entre parejas y la media
> (d2 <- mahalanobis(huswif, colMeans(huswif), cov(huswif)))

       1        2        3        4        5        6        7        8
5.190305 5.933941 6.791552 2.834540 7.482253 5.400837 6.222828 1.150281
       9       10
1.830916 2.162547
```

e) *Realizar un `qqplot` entre los cuantiles de una distribución ji-cuadrado con 5 grados de libertad y las distancias al cuadrado de Mahalanobis. ¿Se ajustan?*

También se puede utilizar la función `chisplot()` de Everitt.

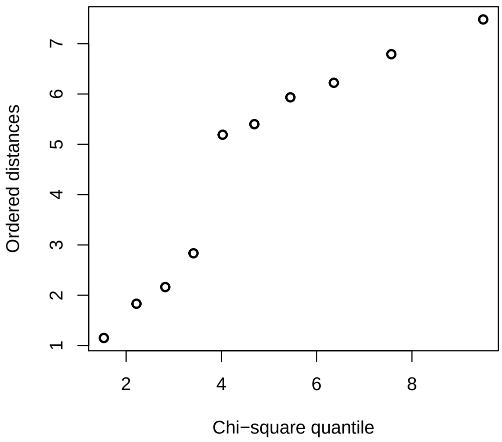
```
> # QQ-plot del cuadrado de las distancias de Mahalobis respecto a
> # la distribución ji-cuadrado con p grados de libertad.
> n <- nrow(huswif)
> p <- ncol(huswif)
> quantiles <- qchisq((1:n)/(n+1),p)
> plot(quantiles, sort(d2),
+      ylab = "Ordered distances",
+      xlab = "Chi-square quantile")
```



No parece que los puntos se ajusten a una recta.

Se repite el mismo gráfico con la función `chisplot()` de Everitt.

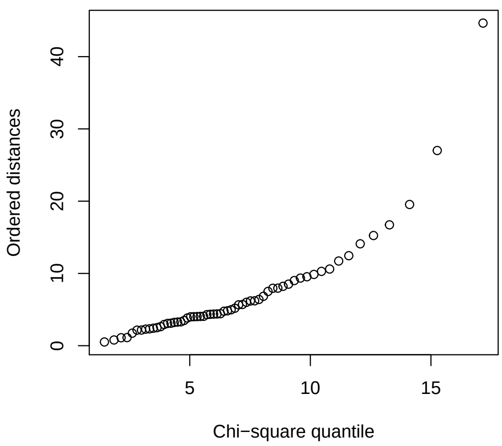
```
> # chisplot function:
> ###############
> chisplot <- function(x) {
+     if (!is.matrix(x)) stop("x is not a matrix")

+     ### determine dimensions
+     n <- nrow(x)
+     p <- ncol(x)
+     #
+     xbar <- apply(x, 2, mean)
+     S <- var(x)
+     S <- solve(S)
+     index <- (1:n)/(n+1)
+     #
+     xcent <- t(t(x) - xbar)
+     di <- apply(xcent, 1, function(x,S) x %*% S %*% x,S)
+     #
+     quant <- qchisq(index,p)
+     plot(quant, sort(di), ylab = "Ordered distances",
+          xlab = "Chi-square quantile", lwd=2,pch=1)

+ }
> ###############
> chisplot(as.matrix(huswif))
```



Ejercicio 5 (∗)

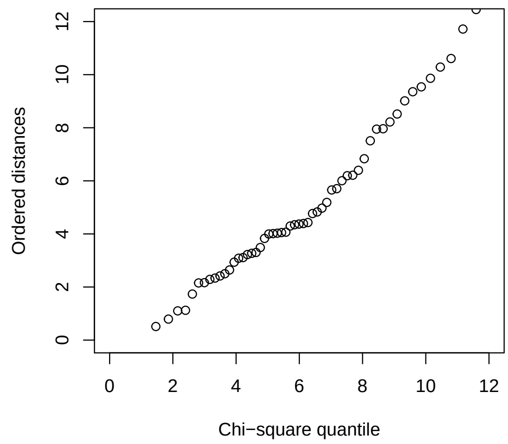
Estudiar el ajuste de las distancias al cuadrado de Mahalanobis para los datos de `airpoll` *del libro de Everitt a una distribución ji-cuadrado.*
Utilizar un `qqplot` *o un* `chisplot` *de Everitt.*

```
> airpoll <- source("data/chap2airpoll.dat")$value
> # QQ-plot del cuadrado de las distancias de Mahalobis respecto a
> # la distribución ji-cuadrado con p grados de libertad.
> n <- nrow(airpoll)
> p <- ncol(airpoll)
> quantiles <- qchisq((1:n)/(n+1),p)
> di <- mahalanobis(airpoll, apply(airpoll,2,mean), cov(airpoll))
> plot(quantiles, sort(di),
+      ylab = "Ordered distances",
+      xlab = "Chi-square quantile")
```



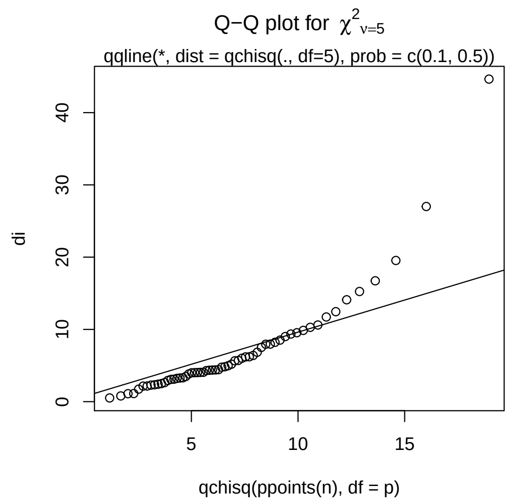
Se observa que los puntos se ajustan a una recta hasta el valor 12 (aprox). Los siguientes valores se pueden considerar atípicos.

```
> plot(quantiles, sort(di),
+       ylab = "Ordered distances", ylim=c(0,12),
+       xlab = "Chi-square quantile", xlim=c(0,12))
```



También podemos utilizar la función `qqplot()`

```
> ## "QQ-Chisquare" : --------------------------
> ## Q-Q plot for Chi^2 data against true theoretical distribution:
> qqplot(qchisq(ppoints(n), df = p), di,
+         main = expression("Q-Q plot for" ~~ {chi^2}[nu == 5]))
> qqline(di, distribution = function(pr) qchisq(pr, df = 5),
+         prob = c(0.1, 0.5))
> mtext("qqline(*, dist = qchisq(., df=5), prob = c(0.1, 0.5))")
```



Ejercicio 6

*Con la base de datos **huswif** del libro de Everitt(2005):*

a) *Realizar una estandarización univariante del tipo*

$$\mathbf{y} = \mathbf{D}^{-1/2}(\mathbf{x} - \bar{\mathbf{x}})$$

donde $\mathbf{D}^{-1/2} = \mathrm{diag}(s_1^{-1}, \ldots, s_p^{-1})$ *y obtener la base de datos transformada.*

```
> restar.la.media <- function(x) x - mean(x)
> D.m12 <- diag(1/apply(huswif,2,sd))
> as.matrix(apply(huswif,2,restar.la.media)) %*% D.m12

          [,1]        [,2]       [,3]        [,4]       [,5]
1   0.76235662  1.16751086  0.3896433  0.18575466 -0.3475997
2  -1.34069612  1.63393218 -0.7792865 -0.27863198 -1.4452832
3  -0.02628816 -1.01883907 -0.6234292  0.65014130  2.0307143
4   1.02523821  0.73024088  1.4806444 -0.58822308 -0.1646525
5   1.55100139 -1.64559271  1.0910011 -2.44576964  0.5671364
6  -0.72730574 -0.49411508 -0.8572152  1.26932348 -0.7134942
7   0.23659343  0.01603323  1.0910011  0.49534575  1.1159781
8   0.58710222  0.16178989  0.3896433  0.03095911 -0.1646525
9  -0.81493294 -0.63987175 -1.1689298  0.49534575 -0.1646525
10 -1.25306892  0.08891156 -1.0130725  0.18575466 -0.7134942

> # que es equivalente a
> apply(huswif,2,scale)

             Hage     Hheight       Wage     Wheight     Hagefm
 [1,]  0.76235662  1.16751086  0.3896433  0.18575466 -0.3475997
 [2,] -1.34069612  1.63393218 -0.7792865 -0.27863198 -1.4452832
 [3,] -0.02628816 -1.01883907 -0.6234292  0.65014130  2.0307143
 [4,]  1.02523821  0.73024088  1.4806444 -0.58822308 -0.1646525
 [5,]  1.55100139 -1.64559271  1.0910011 -2.44576964  0.5671364
 [6,] -0.72730574 -0.49411508 -0.8572152  1.26932348 -0.7134942
 [7,]  0.23659343  0.01603323  1.0910011  0.49534575  1.1159781
 [8,]  0.58710222  0.16178989  0.3896433  0.03095911 -0.1646525
 [9,] -0.81493294 -0.63987175 -1.1689298  0.49534575 -0.1646525
[10,] -1.25306892  0.08891156 -1.0130725  0.18575466 -0.7134942
```

b) *Realizar una estandarización multivariante del tipo*

$$\mathbf{y} = \mathbf{S}^{-1/2}(\mathbf{x} - \bar{\mathbf{x}})$$

```
> # Raiz cuadrada de la inversa de S
> S <- var(huswif)
> diaS <- eigen(S)
> S.m12 <- diaS$vectors %*% diag(1/sqrt(diaS$values)) %*% t(diaS$vectors)
> huswif.mult.scaled <- as.matrix(apply(huswif,2,restar.la.media)) %*% S.m12
> huswif.mult.scaled

          [,1]        [,2]        [,3]        [,4]        [,5]
1   1.83524533  1.22634927 -0.53172937  0.16783296  0.08569421
2  -1.34812083  1.69095379 -0.88185281 -0.65604459  0.22164860
3   0.38226133 -0.96927098 -0.81559169  0.70203580  2.13258002
```

```
4   0.45263797  0.76938038  1.31773552 -0.51382867 -0.19304278
5   0.25196941 -1.43889938  0.09567764 -2.29681665 -0.25261062
6  -0.04880701 -0.71443798  0.06890944  1.37414494 -1.73060992
7  -0.79105214 -0.05772100  2.09069678  0.62790182  0.91019747
8   0.93862221  0.15690882  0.11507895  0.10904285 -0.46852442
9  -0.34052670 -0.70348245 -0.92387620  0.46287007 -0.39022247
10 -1.33222958  0.04021953 -0.53504828  0.02286146 -0.31511009
```

c) *Observar que la distancia euclídea entre las filas de datos tras la estandarización multivariante coinciden con la distancia de Mahalanobis de los datos originales.*

```
> # Distancia euclídea entre observaciones estandarizadas
> round(dist(huswif.mult.scaled),3)

       1     2     3     4     5     6     7     8     9
2  3.342
3  3.389 3.946
4  2.466 3.020 3.803
5  4.024 4.029 3.969 3.097
6  3.526 4.039 4.051 3.153 4.034
7  4.039 3.785 3.486 2.313 4.109 3.550
8  1.636 3.105 3.092 1.588 2.975 2.220 3.018
9  2.988 2.894 2.651 2.969 3.093 1.923 3.380 1.894
10 3.409 1.896 3.239 2.730 3.237 2.532 3.011 2.372 1.374

> # Distancia de Mahalanobis entre observaciones
> round(sqrt(D2.dist(huswif, cov(huswif))),3)

       1     2     3     4     5     6     7     8     9
2  3.342
3  3.389 3.946
4  2.466 3.020 3.803
5  4.024 4.029 3.969 3.097
6  3.526 4.039 4.051 3.153 4.034
7  4.039 3.785 3.486 2.313 4.109 3.550
8  1.636 3.105 3.092 1.588 2.975 2.220 3.018
9  2.988 2.894 2.651 2.969 3.093 1.923 3.380 1.894
10 3.409 1.896 3.239 2.730 3.237 2.532 3.011 2.372 1.374
```

Ejercicio 7

Comprobar que la desviación típica generalizada en el caso de dos variables es

$$|\mathbf{S}|^{1/2} = s_x s_y \sqrt{1 - r^2}$$

donde s_x^2 *y* s_y^2 *son las varianzas muestrales de las dos variables y* r *el coeficiente de correlación.*

El determinante de la matriz de varianzas-covarianzas es

$$|\mathbf{S}| = s_x^2 s_y^2 - s_{xy} s_{yx} = s_x^2 s_y^2 (1 - s_{xy}^2/(s_x^2 s_y^2)) = s_x^2 s_y^2 (1 - r^2)$$

de modo que

$$|\mathbf{S}|^{1/2} = \sqrt{s_x^2 s_y^2 (1 - r^2)} = s_x s_y \sqrt{1 - r^2}$$

Ejercicio 8

Medidas de dependencia lineal con las variables numéricas de la base de datos **crabs** *del paquete MASS de* ***R****, únicamente para los datos de la especie Blue y sexo Macho:*

a) *Hallar la correlación entre las variables CL i CW.*

Calcular la matriz de correlaciones $\mathbf{R}$ *de todas las variables numéricas.*

```
> crabs5BM <- crabs5[crabs$sp=="B" & crabs$sex=="M",]
> with(crabs5BM, cor(CL,CW))

[1] 0.9988421

> cor5BM <- cor(crabs5BM)
> round(cor5BM, 3)

      FL    RW    CL    CW    BD
FL 1.000 0.968 0.995 0.995 0.992
RW 0.968 1.000 0.977 0.979 0.969
CL 0.995 0.977 1.000 0.999 0.994
CW 0.995 0.979 0.999 1.000 0.995
BD 0.992 0.969 0.994 0.995 1.000
```

b) *Hallar la correlación múltiple entre la variable BD y el resto.*

```
> cor.yx <- cor5BM[5,1:4]
> cor.yy <- cor5BM[1:4,1:4]
> as.numeric(sqrt(t(cor.yx) %*% solve(cor.yy) %*% cor.yx))

[1] 0.9952496

> # por regresión
> g <- lm(BD ~ ., data=crabs5BM)
> sqrt(summary(g)$r.squared)

[1] 0.9952496
```

c) *El coeficiente de correlación parcial es una medida de dependencia directa entre dos variables.*

Se sabe que $r_{ij.k_1\dots k_q} = -\frac{s^{ij}}{\sqrt{s^{ii}s^{jj}}}$ *donde* s^{ij} *son los elementos de la matriz* $\mathbf{S}^{-1}$*, llamada matriz de precisión. Calcular la matriz de correlaciones parciales*

$$\mathbf{P} = (-1)^{*}\text{diag}(\mathbf{S}^{-1})^{-1/2}\mathbf{S}^{-1}\text{diag}(\mathbf{S}^{-1})^{-1/2}$$

donde $(-1)^{*}$ *es el producto por* -1 *de todos los elementos excepto los de la diagonal. Comprobar que* ***no*** *es la inversa de la matriz de correlaciones* $\mathbf{R}^{-1}$.

```
> # Matriz de precisión
> Sinv <- solve(var(crabs5BM))
> # Raiz cuadrada de la inversa de la diagonal de Sinv
> d.inv.sq <- diag(1/sqrt(diag(Sinv)))
> # Matriz de correlaciones parciales
> signo <- matrix(-1,5,5)
```

```
> diag(signo) <- rep(1,5)
> mcp <- signo * (d.inv.sq  %*% Sinv %*% d.inv.sq)
> rownames(mcp) <- colnames(mcp) <- colnames(crabs5BM)
> mcp

            FL          RW          CL         CW          BD
FL  1.00000000 -0.23792597  0.35115420 0.08167767  0.25227451
RW -0.23792597  1.00000000  0.05645938 0.32057452 -0.12815608
CL  0.35115420  0.05645938  1.00000000 0.78111319 -0.07428026
CW  0.08167767  0.32057452  0.78111319 1.00000000  0.40375506
BD  0.25227451 -0.12815608 -0.07428026 0.40375506  1.00000000

> # Verificación sobre el par FL, RW por regresión lineal
> rx <- resid(lm(FL ~ CL + CW + BD, data=crabs5BM))
> ry <- resid(lm(RW ~ CL + CW + BD, data=crabs5BM))
> cor(rx,ry)

[1] -0.237926

> # Verificación sobre el par FL, CL con la inversa de la matriz de correlaciones
> Rinv <- solve(cor(crabs5BM))
> -Rinv[1,3]/sqrt(Rinv[1,1]*Rinv[3,3])

[1] 0.3511542

> # Inversa de la matriz de correlaciones
> Rinv

          FL         RW          CL         CW         BD
FL 125.48673  13.588059  -87.385596  -21.96682 -29.027372
RW  13.58806  25.991622   -6.394343  -39.23833   6.711064
CL -87.38560  -6.394343  493.498881 -416.60256  16.949307
CW -21.96682 -39.238335 -416.602560  576.40767 -99.567661
BD -29.02737   6.711064   16.949307  -99.56766 105.504482
```

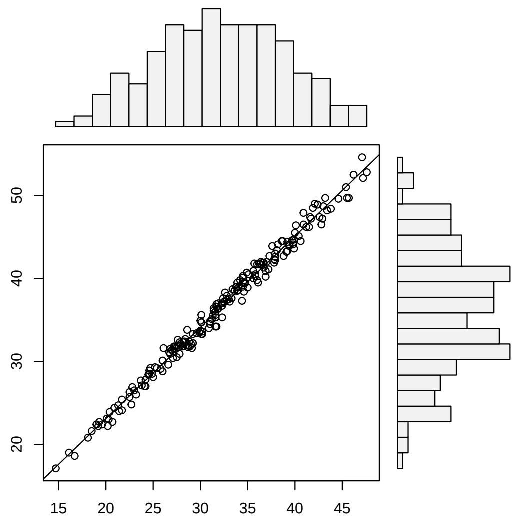
2. Gráficos

Ejercicio 9

Con las variables CL y CW de la base de datos crabs *del paquete MASS de* ***R****:*

a) *Crear el diagrama de dispersión de las dos variables y sus histogramas marginales.*

```
> library(UsingR)
> simple.scatterplot(crabs$CL,crabs$CW)
```

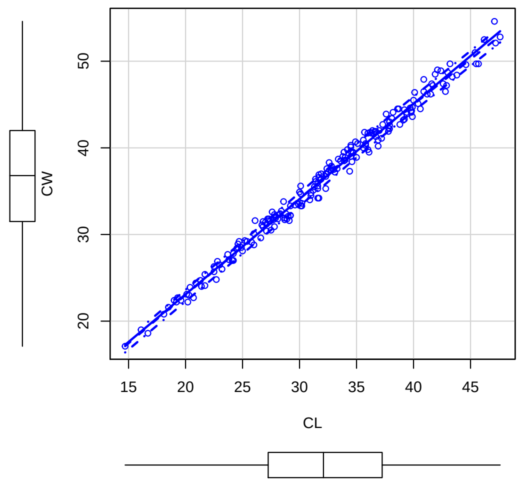


Si se ejecuta la función `simple.scatterplot` en la consola (sin paréntesis), veremos el código de la función. Está claro que habría que eliminar la instrucción

`layout.show(nf)`

b) *Crear el diagrama de dispersión de las dos variables y sus diagramas de caja marginales.*

```
> library(car)
> scatterplot(crabs$CL,crabs$CW, xlab="CL", ylab="CW")
```



También se puede sustituir en la función `simple.scatterplot` los `barplot()` por `boxplot()`.

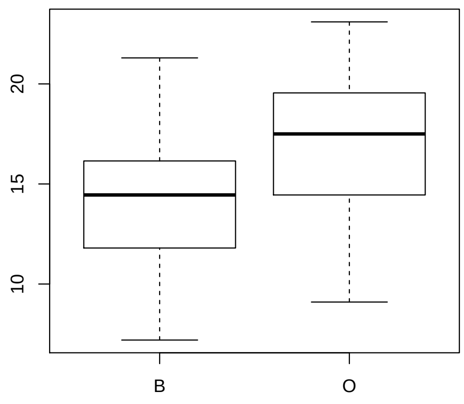
Ejercicio 10

*Con la base de datos **crabs** del paquete **MASS** de **R**, realizar los siguientes diagramas de caja múltiples:*

a) *Para comparar las variables según especie.*

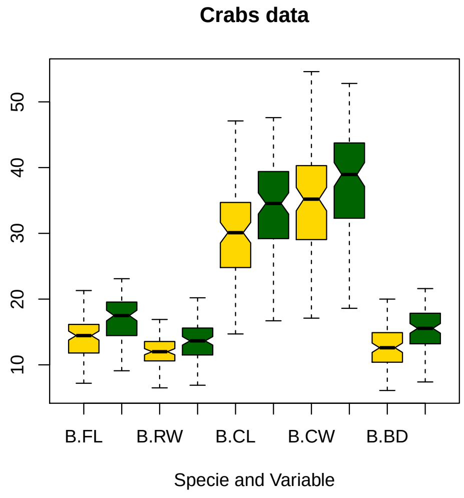
Se puede hacer variable a variable:

```
> boxplot(crabs$FL ~ crabs$sp)
```



O todas a la vez:

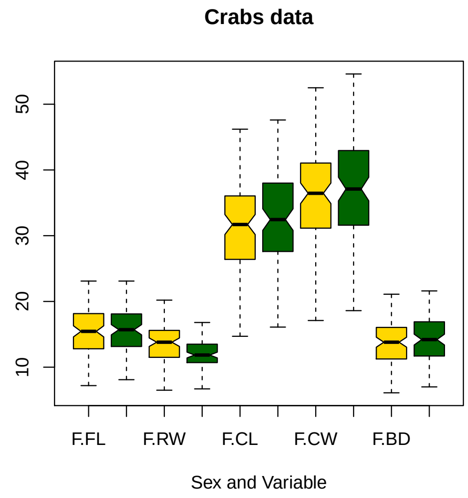
```
> len <- as.vector(as.matrix(crabs5))
> vv <- gl(5,200,labels = colnames(crabs5))
> especie <- rep(crabs$sp,5)
> # Notched Boxplot of Crabs data with 2 Crossed Factors
> # boxes colored for ease of interpretation
> boxplot(len ~ especie*vv, notch=TRUE,
+   col=(c("gold","darkgreen")),
+   main="Crabs data", xlab="Specie and Variable")
```



b) *Para comparar las variables según sexo.*

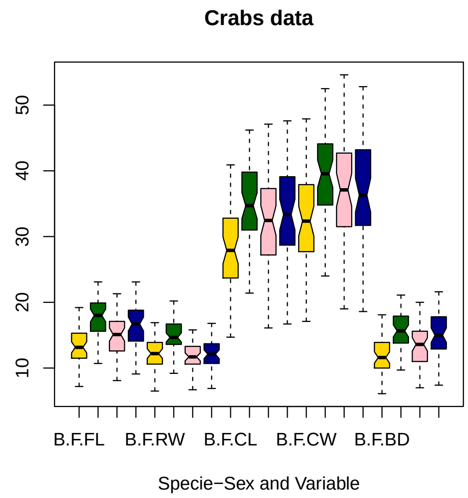
```
> sexo <- rep(crabs$sex,5)
> boxplot(len ~ sexo*vv, notch=TRUE,
```

```
+    col=(c("gold","darkgreen")),
+    main="Crabs data", xlab="Sex and Variable")
```



c) *Para comparar las variables según especie y sexo.*

```
> ii <- interaction(especie,sexo)
> boxplot(len ~ ii*vv, notch=TRUE,
+    col=(c("gold","darkgreen","pink","darkblue")),
+    main="Crabs data", xlab="Specie-Sex and Variable")
```



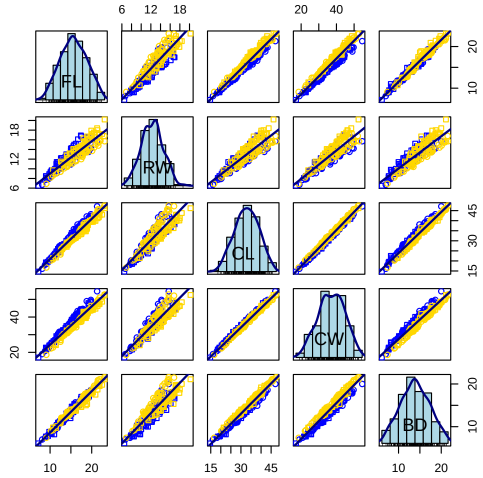
Ejercicio 11

Con la base de datos crabs *del paquete* MASS *de* ***R****, realizar un matriz de diagramas de dispersión según especie y sexo con la instrucción* `pairs()`.

a) *Añadir los histogramas de cada variable a la diagonal.*

b) *Añadir la recta de regresión a cada diagrama de dispersión.*

```
> # Función de panel diagonal: histograma y densidad
> dgp.fn <- function(x,...) {
+   par(new=TRUE)
+   hist(x, col="lightblue", probability=TRUE, axes=FALSE, main="")
+   lines(density(x), col="navy", lwd=2)
+   rug(x)
+ }
> # Función de panel no diagonal: diagrama dispersión y recta de regresión
> pn.fn <- function(x,y,...){
+   points(x,y, col=ifelse(crabs$sp=="B","blue","gold"),
+               pch=ifelse(crabs$sex=="M",21,22))
+   abline(lm(y~x), col="navy", lwd=2);
+ }
>
> # Matriz de diagramas de dispersión
> pairs(crabs5, panel=pn.fn,
+               diag.panel=dgp.fn,
+               label.pos=0.3, cex.labels=1.5)
```

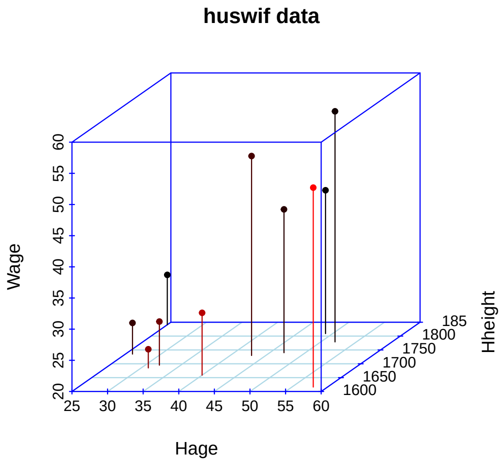


Ejercicio 12

Con la base de datos huswif del libro de Everitt(2005):

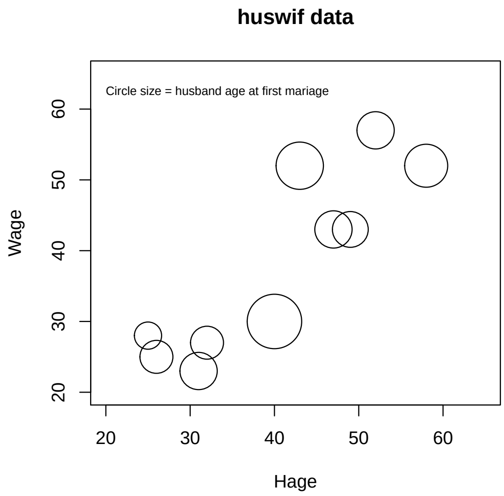
a) *Calcular algún gráfico de dispersión 3D con la función* `scatterplot3d()` *del paquete scatterplot3d.*

```
> library(scatterplot3d)
> with(huswif,
+ scatterplot3d(Hage, Hheight, Wage, highlight.3d=TRUE, col.axis="blue",
+ col.grid="lightblue", main="huswif data", pch=20,type="h"))
```



b) *Realizar un gráfico de dispersión en dos dimensiones de las variables edad del marido y edad de la mujer. Añadir con la función* `symbols()` *la variable edad del marido en el primer matrimonio.*

```
> with(huswif,
+ plot(Hage, Wage, type="n", main="huswif data",
+                  xlim=c(20,65),ylim=c(20,65)))
> with(huswif,
+ symbols(Hage, Wage, circles=Hagefm, inches=0.25, add=TRUE))
> text(20,62,"Circle size = husband age at first mariage",
+      adj=c(0,0),cex=0.7)
```

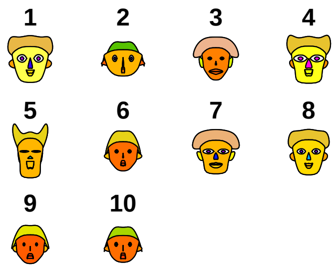


Ejercicio 13 (∗)

Representar con caras de Chernoff las parejas de la base de datos huswif *del libro de Everitt(2005).*[3]

[3] Investigar la función `faces()` del paquete `aplpack`.

```
> library(aplpack)
> faces(huswif)

effect of variables:
 modified item       Var
 "height of face   " "Hage"
 "width of face    " "Hheight"
 "structure of face" "Wage"
 "height of mouth  " "Wheight"
 "width of mouth   " "Hagefm"
 "smiling          " "Hage"
 "height of eyes   " "Hheight"
 "width of eyes    " "Wage"
 "height of hair   " "Wheight"
 "width of hair   "  "Hagefm"
 "style of hair   "  "Hage"
 "height of nose  "  "Hheight"
 "width of nose   "  "Wage"
 "width of ear    "  "Wheight"
 "height of ear   "  "Hagefm"
```



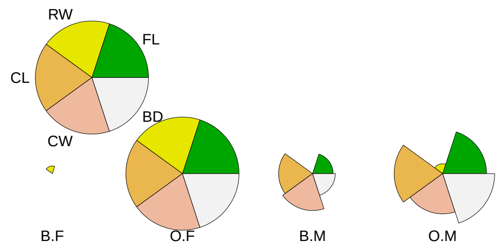
Ejercicio 14 (∗)

*Representar con un gráfico de estrellas los vectores de medias de todas las variables numéricas de la base de datos crabs del paquete MASS de **R** en los cuatro grupos, según especie y sexo.*[4]

```
> attach(crabs)

The following objects are masked from crabs (pos = 22):
    BD, CL, CW, FL, index, RW, sex, sp

> tabla <- aggregate(crabs5, list(sp,sex), mean)
> medias <- as.matrix(tabla[,3:7])
> rownames(medias) <- c("B.F","O.F","B.M","O.M")
> medias
```

[4]Consultar la función `stars()`.

```
        FL     RW     CL     CW     BD
B.F 13.270 12.138 28.102 32.624 11.816
O.F 17.594 14.836 34.618 39.036 15.632
B.M 14.842 11.718 32.014 36.810 13.350
O.M 16.626 12.262 33.688 37.188 15.324

> stars(medias, nrow=1, draw.segments = TRUE, col.segments = terrain.colors(5),
+       mar=c(0,0,0,0), key.loc = c(3,4))
> title("Crabs data by specie and sex", line=-18)
```



Crabs data by specie and sex

Ejercicio 15

Señalar los datos atípicos en la base de datos `crabs`*, únicamente para los datos de la especie Blue y sexo Macho:*

a) *Según un criterio univariante como*

$$\frac{|x_i - \mathrm{med}(x)|}{\mathrm{MAD}(x)} > 5$$

donde $\mathrm{med}(x)$ *es la mediana de las observaciones y* $\mathrm{MAD}(x)$ *la mediana de las desviaciones en valor absoluto respecto a la mediana.*

```
> out.univ1 <- function(x){
+   mediana <- median(x)
+   MAD <- mad(x)
+   sum(abs(x-mediana)/MAD > 5)
+ }
> apply(crabs5BM,2,out.univ1)

FL RW CL CW BD
 0  0  0  0  0
```

No hay valores superiores a 5.

b) *Con otro criterio univariante como*

$$x_i < Q_1 - 1.5 \cdot \mathrm{IQR} \quad \text{o} \quad x_i > Q_3 + 1.5 \cdot \mathrm{IQR}$$

donde Q_1,Q_3 son los cuartiles y IQR *el rango intercuartílico.*

```
> out.univ2 <- function(x){
+   lim.min <- as.numeric(quantile(x, probs = 0.25)) - 1.5*IQR(x)
+   lim.max <- as.numeric(quantile(x, probs = 0.75)) + 1.5*IQR(x)
+   sum( x < lim.min | x > lim.max)
+ }
> apply(crabs5BM,2,out.univ2)

FL RW CL CW BD
 0  1  0  0  0

> y <- crabs5BM$RW
> which(y < as.numeric(quantile(y, probs = 0.25)) - 1.5*IQR(y))

[1] 1

> which(y > as.numeric(quantile(y, probs = 0.75)) + 1.5*IQR(y))

integer(0)
```

Un único dato de `crabs5BM` tiene un valor atípico en `RW`.

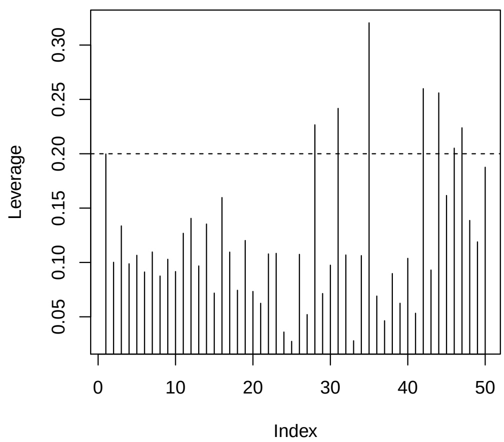
c) *Con un criterio multivariante sencillo como*

1) *Se busca el 50 % de los datos con menor distancia de Mahalanobis al centro $\bar{\mathbf{x}}$, media de todos los datos.*
2) *Se calculan la media $\bar{\mathbf{x}}_R$ y la matriz de covarianzas $\mathbf{S}_R$ con ese conjunto reducido de datos.*
3) *Se calculan las distancias de Mahalanobis d_M^2 de todos los datos a la media $\bar{\mathbf{x}}_R$ con la matriz de covarianzas $\mathbf{S}_R$.*
4) *Se consideran atípicos los datos tales que $d_M^2 > p + 3\sqrt{2p}$, donde p es el número de variables.*

```
> # distancias de mahalanobis al centro
> d2M <- mahalanobis(crabs5BM, colMeans(crabs5BM), cov(crabs5BM))
> # 50% de los datos con menor distancia
> aux <- crabs5BM[d2M < median(d2M),]
> dim(crabs5BM)

[1] 50  5

> dim(aux)

[1] 25  5

> # distancias de mahalanobis al centro reducido
> p <- ncol(crabs5BM)
> d2aux <- mahalanobis(crabs5BM, colMeans(aux), cov(aux))
> which(d2aux > p +3*sqrt(2*p))

16 28 31 35 42 44 46 47
16 28 31 35 42 44 46 47
```

Otro criterio es el de Kutner (2005) que se utiliza en regresión: son outliers los puntos con nivel (leverage) superior al doble del promedio.

```
> # Puntos con un nivel superior al doble del promedio (Kutner, 2005)
> p <- ncol(crabs5BM)
> n <- nrow(crabs5BM)
> fit <- lm(rnorm(n) ~ .,data=crabs5BM) # modelo lineal auxiliar
> which(hatvalues(fit) > 2*p/n)

28 31 35 42 44 46 47
28 31 35 42 44 46 47

> plot(hatvalues(fit), type="h", ylab="Leverage")
> abline(h=2*p/n, lty=2)
```

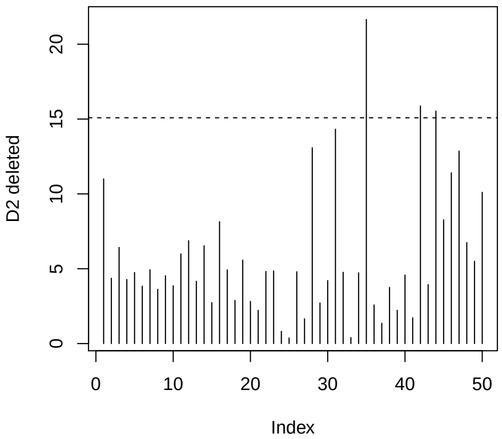


También se puede calcular la distancia de Mahalanobis con el punto eliminado para calcular las medias y la covarianza.

```
> d2del <- numeric(n)
> for (i in 1:n){
+   d2del[i] <- mahalanobis(crabs5BM[i,], colMeans(crabs5BM[-i,]), cov(crabs5BM[-i,]))
+ }
```

La distancia crítica para $\alpha = 0.01$ es

```
> alpha <- 0.01
> d2del.cv <- qchisq(1-alpha,p); d2del.cv

[1] 15.08627

> # Puntos de distancia superior a la crítica
> which( d2del > d2del.cv)

[1] 35 42 44

> plot(d2del, type="h", ylab="D2 deleted")
> abline(h=d2del.cv, lty=2)
```

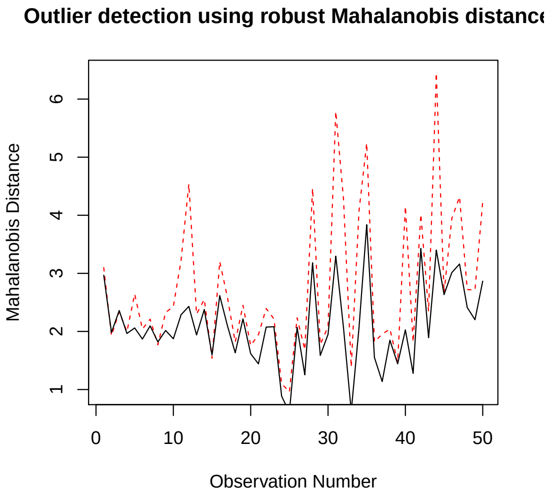


d) *También podemos utilizar la distancia de Mahalanobis d_M^2 calculada con estimadores robustos como los que se obtienen con la función* `cov.rob` *del paquete* `MASS`*. Uno de los métodos más utilizados es el minimum covariance determinant o MCD que calcula los estimadores de centro y covarianza con un grupo de datos reducido, precisamente el que minimiza el determinante de la covarianza (una de las medidas de variabilidad multivariante).*

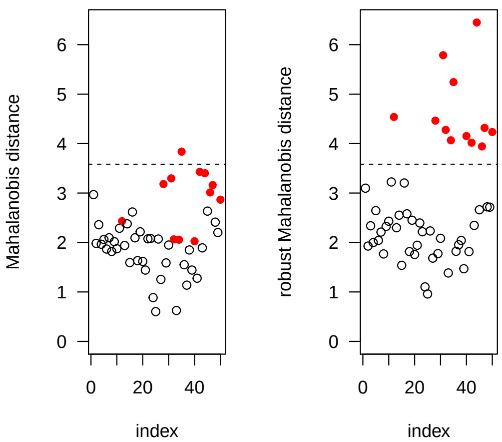
Con estas distancias robustas, podemos dibujar un gráfico para observar por inspección los puntos más extremos. Algunos autores sugieren como punto de corte la cola superior al 97.5 % de la distribución ji-cuadrado asociada a la distancia de Mahalanobis.

Otro gráfico con estas distancias robustas es el ajuste a la distribución ji-cuadrado para observar los valores máximos.

```
> d2M <- mahalanobis(crabs5BM, colMeans(crabs5BM), cov(crabs5BM))
> set.seed(123)
> cov_mcd <- cov.rob(crabs5BM, method="mcd")
> d2M.rob <- mahalanobis(crabs5BM, center=cov_mcd$center, cov=cov_mcd$cov)
> plot(sqrt(d2M.rob), col="red", typ="l", ylab="Mahalanobis Distance",
+      xlab="Observation Number",lty=2)
> lines(sqrt(d2M), typ="l")
> title("Outlier detection using robust Mahalanobis distances")
```



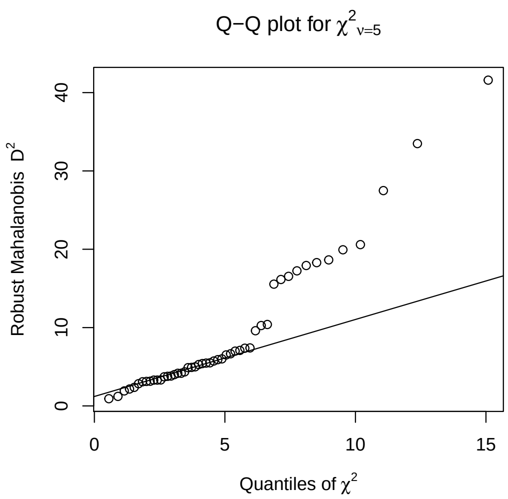

```
> maha1 <- sqrt(d2M)
> maha2 <- sqrt(d2M.rob)
> max.maha <- max(c(maha1, maha2))
> out.id <- ifelse(maha2 <= sqrt(qchisq(0.975, 5)), 0, 1)
> par(mfrow = c(1, 2), las = 1)
> plot(maha1, xlab = "index" ,ylab = "Mahalanobis distance",
+              ylim = c(0, max.maha), col = out.id + 1, pch = 15 * out.id + 1)
> abline(h = sqrt(qchisq(0.975, 5)),lty=2)
> plot(maha2, xlab = "index", ylab = "robust Mahalanobis distance",
+              ylim = c(0, max.maha), col = out.id + 1, pch = 15 * out.id + 1)
> abline(h = sqrt(qchisq(0.975, 5)),lty=2)
> par(mfrow = c(1, 1))
```



Observamos que los individuos atípicos que se detectan con las distancias de Mahalanobis con los estimadores robustos no son detectados con los estimadores habituales.

El gráfico de cuantiles para el ajuste a la distribución ji-cuadrado también nos permite identificar los valores atípicos extremos.

```
> ## Q-Q plot for Chi^2 data against true theoretical distribution:
> n <- nrow(crabs5BM)
> p <- ncol(crabs5BM)
> qqplot(qchisq(ppoints(n), df = p), d2M.rob,
+        xlab = expression("Quantiles of" ~ {chi^2}),
+        ylab = expression("Robust Mahalanobis " ~ {D^2}),
+        main = expression("Q-Q plot for" ~ {chi^2}[nu == 5]))
> qqline(d2M.rob, distribution = function(pr) qchisq(pr, df = 5),
+        prob = c(0.1, 0.5))
```



Tanto en el gráfico de las distancias (derecha), como en éste, observamos un grupo de outliers entre los que destacan 3 de ellos.

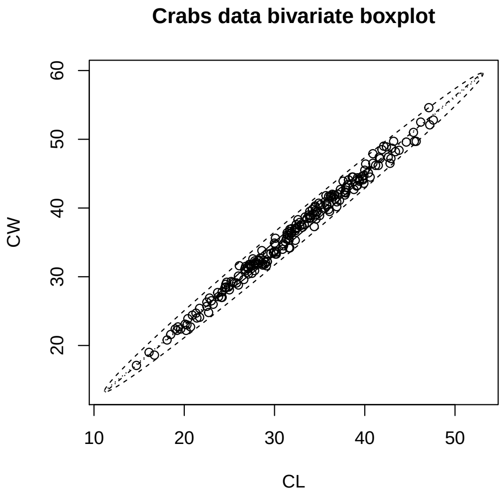
```
> order(d2M.rob,decreasing = T)[1:3]

[1] 44 31 35
```

Otros criterios pueden verse en Peña(2002) 4.5.3 y 11.3.

Ejercicio 16

Realizar un boxplot bivariante con la función `bvbox()` *de Everitt para las variables CL y CW de la base de datos crabs del paquete MASS de* ***R****.*
¿Cuales son las medidas de dispersión robustas que se utilizan en el gráfico?

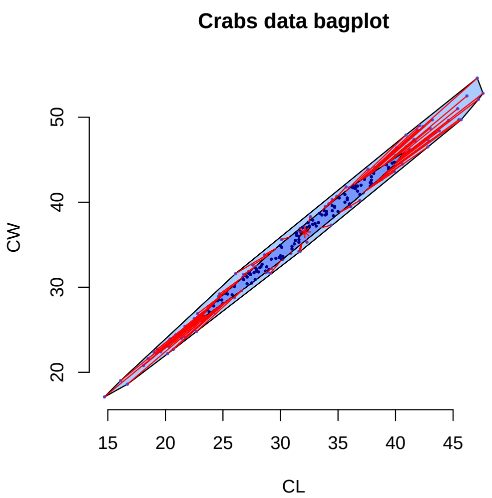
```
> library(MVA)
> bvbox(as.matrix(crabs[,6:7]), xlab="CL", ylab="CW")
> title(main="Crabs data bivariate boxplot")
```



Si se investiga un poco en las funciones del paquete MVA se ve que el cálculo de las medidas robustas lo hace la función `biweight()` y ésta utiliza las medianas, las MAD y una medida de correlación robusta[5].

Otra posibilidad es la función `bagplot()` del paquete `aplpack` que proporciona también una versión bivariante del boxplot. El "saco" contiene el 50 % de los puntos. Además muestra los outliers.

```
> require(aplpack)
> bagplot(crabs$CL,crabs$CW, xlab="CL", ylab="CW",
+         main="Crabs data bagplot")
```



Ejercicio 17 (∗)

Un grupo muy importante de técnicas de representación de datos que no hemos explicado en los documentos de esta asignatura son los ***gráficos en panel*** *(trellis) que podemos hallar en el paquete* `lattice` *de* ***R****.*
Uno de los lugares donde informarse sobre los gráficos trellis es:

`http://www.stat.auckland.ac.nz/~ihaka/787/lectures-trellis.pdf`

[5]Ver `http://people.stat.sc.edu/Hitchcock/chapter2_R_examples.txt`

Escribir un breve informe (menos de una página) que describa las características de los gráficos trellis y discutir sus méritos.
¿Qué hace diferentes a los gráficos trellis de otras técnicas de representación de datos multivariantes? ¿Pueden ser útiles para ti? ¿Cuando crees que es mejor usar gráficos trellis que otras técnicas?
Mejor si se incluyen referencias a las fuentes.

Se valorará el informe.

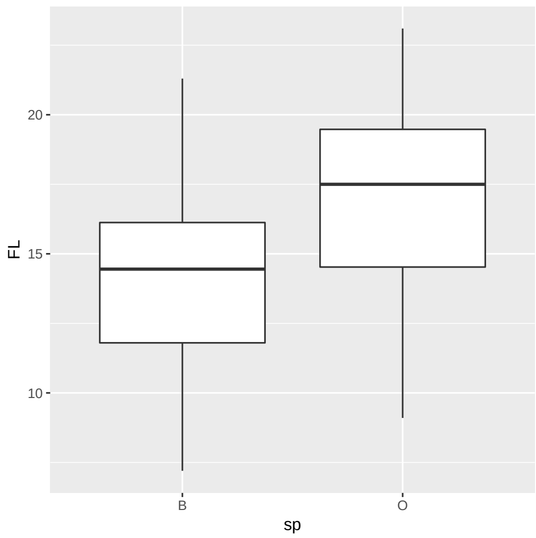
Ejercicio 18 (∗)

*Con la base de datos **crabs** del paquete MASS de **R**, realizar los siguientes diagramas de caja múltiples:*

a) *Para comparar las variables según especie.*

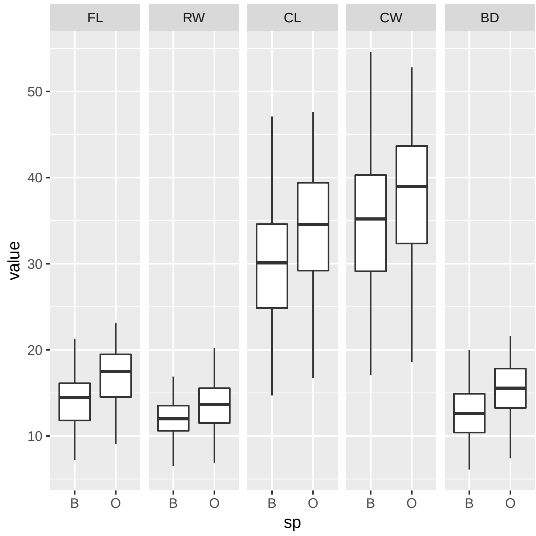
   Se puede hacer variable a variable:

```
> library(ggplot2)
> ggplot(crabs, aes(x=sp, y=FL)) + geom_boxplot()
> p

[1] 5
```



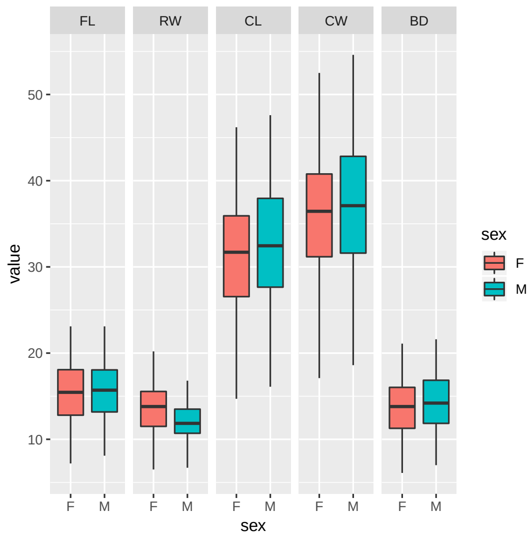
   O todas a la vez:

```
> library(reshape2)
> df <- melt(crabs[,-c(2,3)],"sp")
> ggplot(df, aes(x=sp, y=value)) + geom_boxplot() + facet_grid(~ variable)
```



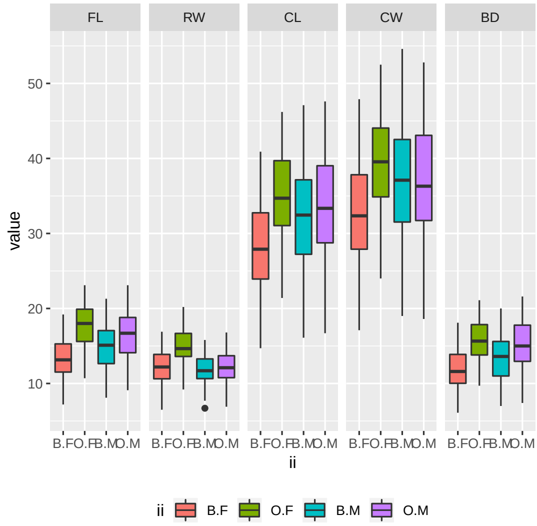
b) *Para comparar las variables según sexo.*

```
> df <- melt(crabs[,-c(1,3)],"sex")
> ggplot(df, aes(x=sex, y=value, fill=sex)) +
+        geom_boxplot() + facet_grid(~ variable)
```



c) *Para comparar las variables según especie y sexo.*

```
> ii <- interaction(crabs$sp,crabs$sex)
> df <- cbind(crabs[,4:8],ii)
> df <- melt(df,"ii")
> ggplot(df, aes(x=ii, y=value, fill=ii)) +
+        geom_boxplot() + facet_grid(~ variable) + theme(legend.position="bottom")
```


FL
RW
CL
CW
BD
value
50
40
30
20
10
B.F O.F B.M O.M
ii
ii B.F O.F B.M O.M