Reporte de investigación

Karina Itzel Rodríguez Conde

2022-05-25

ANÁLISIS DE COMPONENTES PRINCIPALES Y ÁRBOLES DE DECISIÓN

Introducción

El **análisis de componentes principales**, también conocido como *Principal Component Analysis* o PCA, es uno de los algoritmos de selección de características más habituales. Este análisis consiste en una técnica de selección de características concreta que utiliza una transformación ortogonal para convertir un conjunto de observaciones de variables, posiblemente correlacionadas, en un conjunto más reducido de variables que ya no guardan correlación y que se conocen como *componentes principales*.

Su objetivo principal es representar adecuadamente la información contenida en una matriz $n\ x\ p$ con un número menor de variables construidas a partir de combinaciones lineales de las originales. Una de sus ventajas es que permite una representación óptima, en un espacio de dimensión reducida, de las observaciones originales.

Por otra parte, un **árbol de decisión** es un esquema en el que se encuentran todas las posibles consecuencias lógicas de realizar una secuencia de acciones. Se basa en los principios de clasificación que predicen el resultado de una decisión, dando lugar a diferentes ramas de un árbol. Parte de una raíz, que gradualmente tiene diferentes nodos de decisión. La estructura tiene nodos de terminación al final.

Descripción de la matriz de datos

La base de datos utilizada es obtenida del repositorio de R en la paquetería *MASS* y contiene 200 filas y 8 columnas, describiendo 5 medidas morfológicas en 50 cangrejos, cada uno de dos formas de color y ambos sexos, de la especie Leptograpsus variegatus recolectada en Fremantle, W. Australia.

Los datos contienen las siguientes columnas:

1.- **sp:** species - "B" para azul "O" para naranja.

2.- **sex:** F o M.

3.- **index:** dentro de cada uno de los cuatro grupos. 1:50

4.- **FL:** tamaño del lóbulo frontal (mm).

5.- **RW:** anchura trasera (mm).

6.- **CL:** longitud del caparazón (mm).

7.- **CW:** ancho del caparazón (mm).

8.- **BD:** profundidad corporal (mm).

Exploración de la matriz de datos

Paquetería a utilizar

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

1.- Cargando la base de datos

```
data(crabs, package = "MASS")
base <- crabs
head(base)
```

```
##   sp sex index   FL  RW   CL   CW  BD
## 1  B   M     1  8.1 6.7 16.1 19.0 7.0
## 2  B   M     2  8.8 7.7 18.1 20.8 7.4
## 3  B   M     3  9.2 7.8 19.0 22.4 7.7
## 4  B   M     4  9.6 7.9 20.1 23.1 8.2
## 5  B   M     5  9.8 8.0 20.3 23.0 8.2
## 6  B   M     6 10.8 9.0 23.0 26.5 9.8
```

2.- Dimensión de la base

```
dim(base)
```

```
## [1] 200   8
```

La base de datos contiene 200 observaciones y 8 variables.

3.- Nombre de las variables

```
colnames(base)
```

```
## [1] "sp"    "sex"   "index" "FL"    "RW"    "CL"    "CW"    "BD"
```

4.- Tipo de variables

```
str(base)
```

```
## '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 ...
```

5.- Presencia de NA

```
anyNA(base)
```

```
## [1] FALSE
```

Esta base no contiene datos nulos.

Tratamiento de la matriz

Paquetería y librería a utilizar

```
install.packages("tidyverse")
library(tidyverse)
```

1.- Cambio de etiquetas

```
base <- base %>% rename("Especies" = sp, "Sexo" = sex, "índice" = index,
                        "Tamaño del lóbulo frontal" = FL, "Anchura trasera" = RW,
                        "Longitud del caparazón" = CL,
                        "Ancho del caparazón" = CW, "Profundidad corporal" = BD)
base <- base %>% mutate(Especies =  recode(Especies, 'B' = "Blue",
                                           'O' = "Orange"))
base <- base %>% mutate(Sexo =  recode(Sexo, 'M' = "Macho",
                                       'F' = "Hembra"))
head(base)
```

```
##   Especies  Sexo índice Tamaño del lóbulo frontal Anchura trasera
## 1     Blue Macho      1                      8.1             6.7
## 2     Blue Macho      2                      8.8             7.7
## 3     Blue Macho      3                      9.2             7.8
## 4     Blue Macho      4                      9.6             7.9
## 5     Blue Macho      5                      9.8             8.0
## 6     Blue Macho      6                     10.8             9.0
##   Longitud del caparazón Ancho del caparazón Profundidad corporal
## 1                   16.1                19.0                   7.0
## 2                   18.1                20.8                   7.4
## 3                   19.0                22.4                   7.7
## 4                   20.1                23.1                   8.2
## 5                   20.3                23.0                   8.2
## 6                   23.0                26.5                   9.8
```

2.- Resumen de los datos

```
summary(base)
```

```
##    Especies       Sexo         índice     Tamaño del lóbulo frontal
##  Blue  :100   Hembra:100   Min.   : 1.0   Min.   : 7.20
##  Orange:100   Macho :100   1st Qu.:13.0   1st Qu.:12.90
##                            Median :25.5   Median :15.55
##                            Mean   :25.5   Mean   :15.58
##                            3rd Qu.:38.0   3rd Qu.:18.05
##                            Max.   :50.0   Max.   :23.10
##  Anchura trasera Longitud del caparazón Ancho del caparazón
##  Min.   : 6.50   Min.   :14.70          Min.   :17.10
##  1st Qu.:11.00   1st Qu.:27.27          1st Qu.:31.50
##  Median :12.80   Median :32.10          Median :36.80
##  Mean   :12.74   Mean   :32.11          Mean   :36.41
##  3rd Qu.:14.30   3rd Qu.:37.23          3rd Qu.:42.00
##  Max.   :20.20   Max.   :47.60          Max.   :54.60
##  Profundidad corporal
##  Min.   : 6.10
##  1st Qu.:11.40
##  Median :13.90
##  Mean   :14.03
##  3rd Qu.:16.60
##  Max.   :21.60
```

3.- Configuración y/o filtrado de variables

Se genera una nueva matriz *x1* que filtrará las variables cuantitativas de especie naranja, eliminando las variables Especies, Sexo e índice

```
x1 <- base[101:200,-cbind(1,2,3)]
head(x1)
```

```
##     Tamaño del lóbulo frontal Anchura trasera Longitud del caparazón
## 101                       9.1             6.9                   16.7
## 102                      10.2             8.2                   20.2
## 103                      10.7             8.6                   20.7
## 104                      11.4             9.0                   22.7
## 105                      12.5             9.4                   23.2
## 106                      12.5             9.4                   24.2
##     Ancho del caparazón Profundidad corporal
## 101                18.6                  7.4
## 102                22.2                  9.0
## 103                22.7                  9.2
## 104                24.8                 10.1
## 105                26.0                 10.8
## 106                27.0                 11.2
```

METODOLOGÍA DE ANÁLISIS

El primer componente principal será la *combinación lineal* de las variables originales que tengan máxima varianza y los valores de los n individuos se representan mediante el vector **z1**. Al tratarse de las variables originales, su media es cero.

Se calcula la matriz de covarianza muestral y la matriz de correlaciones de las variables originales, donde se obtienen los valores y vectores propios. Mediante ellos, se calcula la proporción de variabilidad y la proporción de variabilidad acumulada y es en este último donde se obtienen aquellos factores que conforman el número de componentes, considerando el 80% de la varianza explicada. La obtención de los coeficientes se realiza mediante la matriz de autovectores.

Para la elaboración de un árbol de decisión, se deja la columna categórica y las variables cuantitativas que sean de interés y se consiguen aquellas variables para elaborar el árbol. Como el árbol es de todos los datos, se escoge un tamaño de muestra que será de entrenamiento y se calcula la probabilidad de predicción y la matriz de confusión para saber con cuántos casos se equivoca.

RESULTADOS

PCA paso a paso

1.- Se transforma la matriz en un data.frame

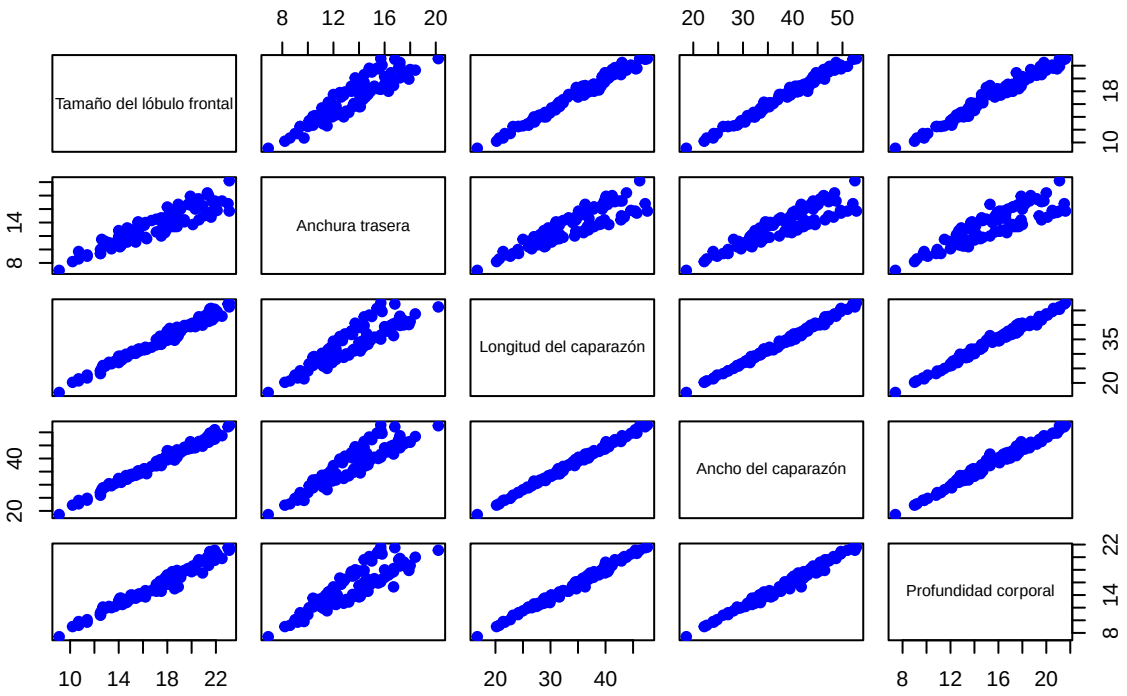
```
x1 <- as.data.frame(x1)
```

2.- Definir n(individuos) y p(variables)

```
n <- dim(x1)[1]
p <- dim(x1)[2]
```

3.- Generación de un scatterplot de las variables originales, sin tomar en cuenta la variable cualitativa (Sexo)

```
pairs(x1,col="blue", pch=19,
      main="Variables originales")
```



4.- Obtención de la media por columna y la **matriz de covarianza muestral**

```
mu <-colMeans(x1)
mu
```

```
## Tamaño del lóbulo frontal           Anchura trasera    Longitud del caparazón
##                    17.110                    13.549                    34.153
##       Ancho del caparazón      Profundidad corporal
##                    38.112                    15.478
```

```
s <-cov(x1)
s
```

```
##                           Tamaño del lóbulo frontal Anchura trasera
## Tamaño del lóbulo frontal                 10.729394        7.718697
## Anchura trasera                            7.718697        6.788787
## Longitud del caparazón                    21.901586       15.418993
## Ancho del caparazón                       24.490889       17.673143
## Profundidad corporal                      10.140828        7.059574
##                           Longitud del caparazón Ancho del caparazón
## Tamaño del lóbulo frontal               21.90159             24.49089
## Anchura trasera                         15.41899             17.67314
## Longitud del caparazón                  45.75524             50.84400
## Ancho del caparazón                     50.84400             56.86551
## Profundidad corporal                    21.19259             23.53077
##                           Profundidad corporal
## Tamaño del lóbulo frontal            10.140828
## Anchura trasera                       7.059574
## Longitud del caparazón               21.192592
## Ancho del caparazón                  23.530772
## Profundidad corporal                  9.931834
```

5.- Obtención de los **valores** y **vectores propios** desde la matriz de covarianza muestral

```
es <- eigen(s)
es
```

```
## eigen() decomposition
## $values
## [1] 128.23538084   1.49544362   0.15038792   0.11626900   0.07328832
##
## $vectors
##            [,1]        [,2]        [,3]        [,4]       [,5]
## [1,] -0.2872625  0.08554827  0.94853974 -0.09923437 -0.0242204
## [2,] -0.2068582  0.92901432 -0.12254943  0.25648753 -0.1154896
## [3,] -0.5961899 -0.29553956 -0.15080959  0.20109108 -0.7028746
## [4,] -0.6655734  0.02380326 -0.24831443 -0.53716176  0.4541395
## [5,] -0.2761513 -0.20421472  0.02916177  0.77161279  0.5346028
```

5.1.- Separación de la matriz de valores propios:

```
eigen.val <-es$values
eigen.val
```

```
## [1] 128.23538084   1.49544362   0.15038792   0.11626900   0.07328832
```

5.2.- Separación de la matriz de vectores propios:

```
eigen.vec <-es$vectors
eigen.vec
```

```
##            [,1]        [,2]        [,3]        [,4]       [,5]
## [1,] -0.2872625  0.08554827  0.94853974 -0.09923437 -0.0242204
## [2,] -0.2068582  0.92901432 -0.12254943  0.25648753 -0.1154896
## [3,] -0.5961899 -0.29553956 -0.15080959  0.20109108 -0.7028746
## [4,] -0.6655734  0.02380326 -0.24831443 -0.53716176  0.4541395
## [5,] -0.2761513 -0.20421472  0.02916177  0.77161279  0.5346028
```

6.- Proporción de variabilidad para cada valor

6.1.- Para la matriz de valores propios:

```
pro.var<-eigen.val/sum(eigen.val)
pro.var
```

```
## [1] 0.9858893058 0.0114971536 0.0011562007 0.0008938903 0.0005634496
```

6.2.- Proporción de variabilidad acumulada:

```
pro.var.acum<-cumsum(eigen.val)/sum(eigen.val)
pro.var.acum
```

```
## [1] 0.9858893 0.9973865 0.9985427 0.9994366 1.0000000
```

7.- Obtención de la matriz de correlaciones

```
R<-cor(x1)
R
```

```
##                           Tamaño del lóbulo frontal Anchura trasera
## Tamaño del lóbulo frontal                 1.0000000       0.9043995
## Anchura trasera                           0.9043995       1.0000000
## Longitud del caparazón                    0.9884792       0.8748619
## Ancho del caparazón                       0.9914995       0.8994836
```

```
## Profundidad corporal                      0.9823610       0.8597410
##                            Longitud del caparazón Ancho del caparazón
## Tamaño del lóbulo frontal              0.9884792           0.9914995
## Anchura trasera                        0.8748619           0.8994836
## Longitud del caparazón                 1.0000000           0.9967697
## Ancho del caparazón                    0.9967697           1.0000000
## Profundidad corporal                   0.9941431           0.9901408
##                            Profundidad corporal
## Tamaño del lóbulo frontal            0.9823610
## Anchura trasera                      0.8597410
## Longitud del caparazón               0.9941431
## Ancho del caparazón                  0.9901408
## Profundidad corporal                 1.0000000
```

8.- Obtención de los *valores* y *vectores propios* a partir de la **matriz de correlaciones**

```
eR<-eigen(R)
eR
```

```
## eigen() decomposition
## $values
## [1] 4.796308210 0.182189340 0.012967173 0.006854286 0.001680991
##
## $vectors
##            [,1]        [,2]        [,3]        [,4]        [,5]
## [1,] -0.4540551 -0.08648822  0.84960538  0.25366033 -0.01344998
## [2,] -0.4224434  0.88814833 -0.15498291  0.06204351 -0.06974959
## [3,] -0.4530579 -0.26767350 -0.15100960 -0.43438217 -0.71526055
## [4,] -0.4551379 -0.13943381 -0.08476682 -0.54187613  0.68745367
## [5,] -0.4505148 -0.33559118 -0.47345716  0.67043973  0.10374913
```

9.- Separación de la matriz de valores y vectores propios

9.1.- Separación de la matriz de valores propios:

```
eigen.val.R<-eR$values
eigen.val.R
```

```
## [1] 4.796308210 0.182189340 0.012967173 0.006854286 0.001680991
```

9.2.- Separación de la matriz de vectores propios:

```
eigen.vec.R<-eR$vectors
eigen.vec.R
```

```
##            [,1]        [,2]        [,3]        [,4]        [,5]
## [1,] -0.4540551 -0.08648822  0.84960538  0.25366033 -0.01344998
## [2,] -0.4224434  0.88814833 -0.15498291  0.06204351 -0.06974959
## [3,] -0.4530579 -0.26767350 -0.15100960 -0.43438217 -0.71526055
## [4,] -0.4551379 -0.13943381 -0.08476682 -0.54187613  0.68745367
## [5,] -0.4505148 -0.33559118 -0.47345716  0.67043973  0.10374913
```

10.- Cálculo de la proporción de variabilidad

10.1.- Para la matriz de valores propios:

```
pro.var.R<-eigen.val/sum(eigen.val)
pro.var.R
```

```
## [1] 0.9858893058 0.0114971536 0.0011562007 0.0008938903 0.0005634496
```

10.2.- Proporción de variabilidad acumulada:

En este punto, se selecciona el número de componentes siguiendo el criterio del 80% de la varianza explicada.

```
pro.var.acum.R<-cumsum(eigen.val)/sum(eigen.val)
pro.var.acum.R
```

```
## [1] 0.9858893 0.9973865 0.9985427 0.9994366 1.0000000
```

En este caso, se selecciona 1 factor (0.985% de varianza explicada)

11.- Cálculo de la media de los valores propios

```
mean(eigen.val.R)
```

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

12.- Obtención de coeficientes

Centrar los datos con respecto a la media.

12.1.- Construcción de matriz de 1:

```
ones<-matrix(rep(1,n),nrow=n, ncol=1)
```

12.2.- Construcción de la matriz centrada:

```
X.cen<-as.matrix(x1)-ones%*%mu
```

13.- Construcción de la matriz diagonal de las covarianzas

```
Dx<-diag(diag(s))
Dx
```

```
##          [,1]     [,2]     [,3]     [,4]     [,5]
## [1,] 10.72939 0.000000  0.00000  0.00000 0.000000
## [2,]  0.00000 6.788787  0.00000  0.00000 0.000000
## [3,]  0.00000 0.000000 45.75524  0.00000 0.000000
## [4,]  0.00000 0.000000  0.00000 56.86551 0.000000
## [5,]  0.00000 0.000000  0.00000  0.00000 9.931834
```

14.- Construcción de la matriz centrada multiplicada por Dxˆ1/2

```
Y<-X.cen%*%solve(Dx)^(1/2)
```

15.- Construcción de los coeficientes o scores eigen.vec.R matriz de autovectores

```
scores<-Y%*%eigen.vec.R
scores
```

```
##            [,1]         [,2]         [,3]         [,4]          [,5]
## 101  5.68976660 -0.143323952  0.140444779  0.025759157  0.0116745322
## 102  4.64608335 -0.104681623 -0.010545280 -0.001169599 -0.0168765935
## 103  4.41966317 -0.031863777  0.048520355  0.021585209 -0.0303424604
## 104  3.86841553 -0.127809531  0.002825538 -0.002554171 -0.0343357660
## 105  3.44509861 -0.137020676  0.134531484  0.122732933  0.0300092091
## 106  3.26058337 -0.237677405  0.040872669  0.071752823  0.0285994806
## 107  2.71286727 -0.102437440 -0.162362290  0.057581689  0.0043966479
## 108  2.36173753  0.005120839 -0.125929566 -0.053749064  0.0191194602
## 109  2.59029823 -0.251067956  0.029681366 -0.012644620  0.0157470187
## 110  2.25960148 -0.025758867 -0.006295544 -0.047901796 -0.0161137452
## 111  1.79182361 -0.057788349 -0.150495209 -0.052627529 -0.0258844753
## 112  1.94336291 -0.458713388 -0.108026626  0.062026582  0.0115771229
```

```
## 113  1.98300642 -0.386247903 -0.056098313 -0.019159213 -0.0436489959
## 114  1.93067362 -0.329089561 -0.092483839  0.032280300  0.0272265524
## 115  2.00227760 -0.319524230  0.001743450 -0.033080505 -0.0019079944
## 116  2.12321987 -0.213733041  0.054926750  0.042035086  0.0110674061
## 117  1.73934527 -0.173837967 -0.146816851  0.043193197 -0.0081834378
## 118  1.65861807 -0.161596371  0.031856351 -0.213114958  0.0116228038
## 119  1.77907528 -0.202855961  0.060449870  0.012084780 -0.0060189309
## 120  1.39352921 -0.276668937 -0.029132978  0.078378241 -0.0035572952
## 121  1.30247747 -0.272576611 -0.043553757 -0.035767040 -0.0197309733
## 122  1.45395365 -0.339154963  0.138268822 -0.033459857  0.0141303572
## 123  0.99727336 -0.117111125  0.005256345  0.025441483 -0.0974175812
## 124  0.81476263 -0.363987065  0.057860846  0.043431324 -0.0354676260
## 125  0.97519416 -0.365464682  0.153733775  0.127861211 -0.0633690345
## 126  0.01922430 -0.394949728 -0.007263623 -0.087160538  0.0150278231
## 127 -0.23615400 -0.442563290 -0.047765542 -0.066550918 -0.0367149790
## 128  0.25499437 -0.514111407  0.223649635 -0.002063851 -0.0661372949
## 129 -0.05264861 -0.388945161  0.044995533  0.092907772  0.0205915876
## 130 -0.32584693 -0.610219348 -0.002426901  0.048514669  0.0240569205
## 131 -0.50264312 -0.511168049 -0.002692744 -0.113603416  0.0114666137
## 132 -0.69409032 -0.406747050 -0.096667666  0.017789697  0.0750369865
## 133 -1.00990154 -0.429402392 -0.140279470 -0.173762322 -0.0249511282
## 134 -0.97000365 -0.545328048 -0.119963589  0.034675516  0.0483044229
## 135 -0.91574146 -0.498509864 -0.002390385 -0.006952542  0.0175400819
## 136 -0.72618370 -0.365428737  0.075933053  0.111561515 -0.0548233598
## 137 -0.88358433 -0.486552760  0.041825657  0.147099831  0.0138172060
## 138 -1.27235797 -0.512621354 -0.111439789 -0.044802293  0.0053541323
## 139 -1.37714032 -0.409747710  0.044720949  0.001139072 -0.0069747741
## 140 -1.55335169 -0.366174760 -0.016139295 -0.094439395  0.0145469622
## 141 -1.61684023 -0.669271010  0.171930257 -0.101371611 -0.0326566510
## 142 -2.29643160 -0.738280898 -0.051988515  0.009403636 -0.0510796788
## 143 -2.46951516 -0.539923200  0.036018020 -0.044603011 -0.0221281412
## 144 -3.15935366 -0.691532339 -0.175531275 -0.021479936 -0.0352041827
## 145 -3.12751179 -0.704228090 -0.103040413 -0.092782004 -0.0639622602
## 146 -2.71417170 -0.736756598  0.075973565  0.042048395  0.0420991872
## 147 -3.34758301 -0.675297391 -0.126104726 -0.030187688  0.0934696520
## 148 -3.16766746 -0.525055276  0.043559126  0.012013820  0.0271835684
## 149 -3.92254334 -0.463547354 -0.018864342 -0.028349995 -0.0173802198
## 150 -3.84139408 -0.880565870  0.040681971 -0.101513211  0.0364657626
## 151  4.03019727  0.227464112 -0.137290516  0.037053907  0.0044564040
## 152  4.00239737  0.035474213  0.081215622  0.031630966 -0.0109310808
## 153  3.21288716  0.002748108  0.052485717  0.028640501  0.0132354688
## 154  2.74338157  0.391567513 -0.133397223  0.062906257 -0.0024650480
## 155  2.57931646  0.188642784 -0.126933648  0.040857900  0.0240725497
## 156  1.99408349  0.233653270 -0.041064626  0.049292741  0.1066471860
## 157  1.65295167  0.440068654 -0.161047242 -0.095451524 -0.0133244958
## 158  1.82033500  0.287716899  0.004873335 -0.040987583  0.0142423356
## 159  1.34853648  0.486489045 -0.062424264 -0.112402933 -0.0137523851
## 160  1.25116015  0.349063652 -0.071048027 -0.063826519 -0.0109988666
## 161  1.26816882  0.036712083 -0.106372175  0.098486988 -0.0166654157
## 162  0.79382606  0.342454206 -0.081939418 -0.005184040 -0.0001824996
## 163  0.71678295  0.525308752 -0.077788633 -0.097157815 -0.0475074654
## 164  0.78904111  0.597931064 -0.061349363  0.001240201 -0.0596702458
## 165  0.83813924  0.419308330 -0.009042895 -0.024479347 -0.0193470237
## 166  0.64148742  0.341518002  0.037776969 -0.075715801  0.0315456967
```

```
## 167  0.64692947  0.392319048  0.050192894 -0.088950735  0.0558411670
## 168  0.62980052  0.514567960  0.089487969 -0.093525074 -0.0259143039
## 169  0.23751520  0.443553709  0.019733522  0.019523178  0.0505319593
## 170  0.09828281  0.476459464  0.106502873 -0.031757594 -0.0261168065
## 171 -0.30631442  0.191459871  0.013683838 -0.055169586  0.0812692871
## 172 -0.34349566  0.194046370 -0.045613132  0.075421668  0.0370627600
## 173 -0.02712916  0.518761514  0.189395495 -0.037605846 -0.0177962201
## 174 -0.16339609  0.135483150  0.094892659  0.028123621  0.0473050233
## 175 -0.49455906  0.366066185  0.089318266  0.013453884  0.0361818658
## 176 -1.36155102  0.492209166 -0.330088739 -0.091100395  0.0287844630
## 177 -0.81017963  0.553927694  0.039281163  0.043285885  0.0755690941
## 178 -0.76633535  0.493869839  0.101620000  0.007540256 -0.0245237938
## 179 -1.02708822  0.443601530 -0.023470679 -0.054096727  0.0372765423
## 180 -0.94882538  0.017524246  0.022192123 -0.090850448  0.0564915671
## 181 -0.40677384  0.290265390  0.375024078 -0.091337825  0.0122652781
## 182 -0.91678354  0.289346846  0.107966430  0.131516655  0.0293425398
## 183 -1.09392208  0.894470842  0.215332354 -0.219918646  0.0025053448
## 184 -1.35926310  0.420398734  0.043263848 -0.041432959 -0.0341096880
## 185 -1.48988899  0.470560904 -0.040278861  0.022826758 -0.0121975240
## 186 -1.97517881  0.386946181 -0.114569592  0.091261058 -0.1127675223
## 187 -1.92842079  0.393763153 -0.003886175  0.051097623 -0.0405685193
## 188 -2.33696825  0.762968883 -0.124942667 -0.142543695  0.0785196183
## 189 -2.06931036  0.384747102  0.010336118 -0.131771665 -0.0465856806
## 190 -2.19235426  0.498937146 -0.104041648  0.189731028 -0.0584709454
## 191 -1.91295461  0.164332524  0.118282069  0.074688434 -0.0046243896
## 192 -2.48010731  0.492943877 -0.128478092  0.242048712  0.0580881833
## 193 -2.63668542  0.418002697 -0.143896069  0.103166215 -0.0371199340
## 194 -2.07541821  0.341288757  0.301376225 -0.048533926 -0.0351060719
## 195 -3.28083344  0.489420328 -0.212130190  0.043208184 -0.0803962523
## 196 -2.73707673  0.632429683  0.115677514  0.089920466 -0.0385278068
## 197 -2.85524503  0.183610768  0.089411247  0.179218917  0.0522543943
## 198 -2.95723822  0.183753096  0.128021181  0.103647212 -0.0314702704
## 199 -3.18854900  0.096098607  0.215033741  0.094830180  0.0521578247
## 200 -4.38764149  0.767542276 -0.117245014  0.010734433  0.0202297835
```

16.- Se nombran las columnas PC1...PC5

```
colnames(scores)<-c("PC1","PC2","PC3","PC4", "PC5")
```

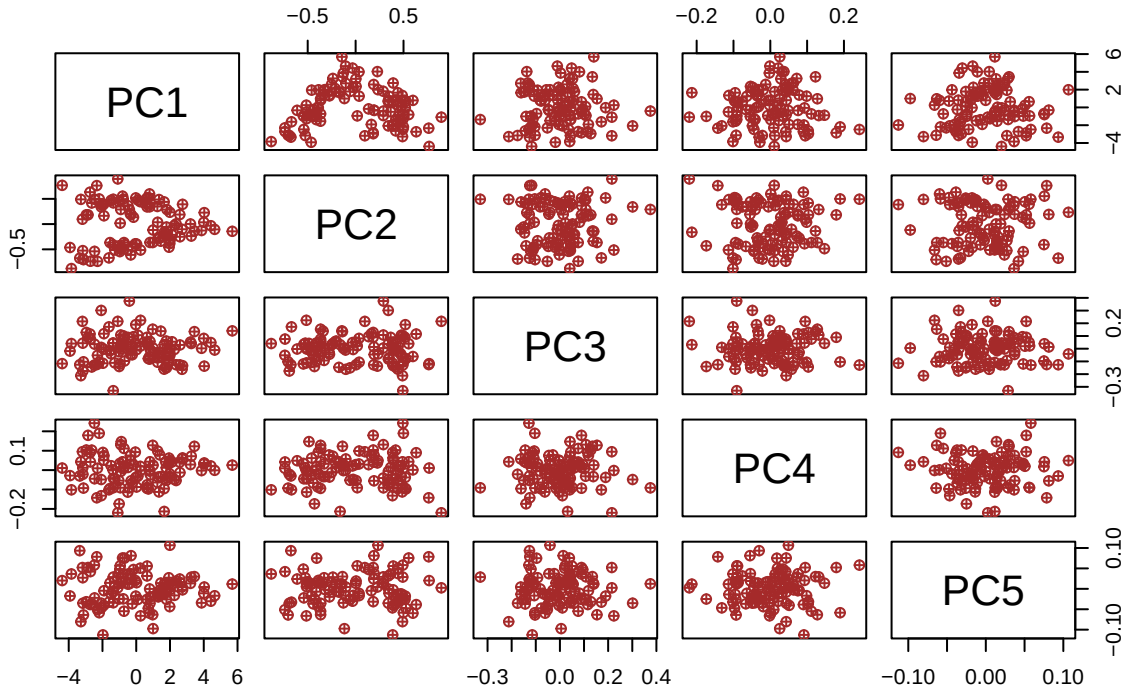
17.- Visualización de los scores

```
scores[1:10,]
```

```
##          PC1          PC2          PC3          PC4          PC5
## 101 5.689767 -0.143323952  0.140444779  0.025759157  0.011674532
## 102 4.646083 -0.104681623 -0.010545280 -0.001169599 -0.016876593
## 103 4.419663 -0.031863777  0.048520355  0.021585209 -0.030342460
## 104 3.868416 -0.127809531  0.002825538 -0.002554171 -0.034335766
## 105 3.445099 -0.137020676  0.134531484  0.122732933  0.030009209
## 106 3.260583 -0.237677405  0.040872669  0.071752823  0.028599481
## 107 2.712867 -0.102437440 -0.162362290  0.057581689  0.004396648
## 108 2.361738  0.005120839 -0.125929566 -0.053749064  0.019119460
## 109 2.590298 -0.251067956  0.029681366 -0.012644620  0.015747019
## 110 2.259601 -0.025758867 -0.006295544 -0.047901796 -0.016113745
```

18.- Generación del gráfico de los scores

```
pairs(scores, main="scores", col="brown", pch=10)
```



Análisis de componentes principales vía sintetizada

1.- Cálculo de la varianza a las columnas: 1 = filas, 2 = columnas

```
apply(x1, 2, var)
```

```
## Tamaño del lóbulo frontal           Anchura trasera    Longitud del caparazón
##                 10.729394                  6.788787                 45.755243
##       Ancho del caparazón      Profundidad corporal
##                 56.865511                  9.931834
```

2.- Aplicación de la función **prcomp** para reducir la dimensionalidad y centrado por la media y escalada por la desviacion standar (dividir entre sd).

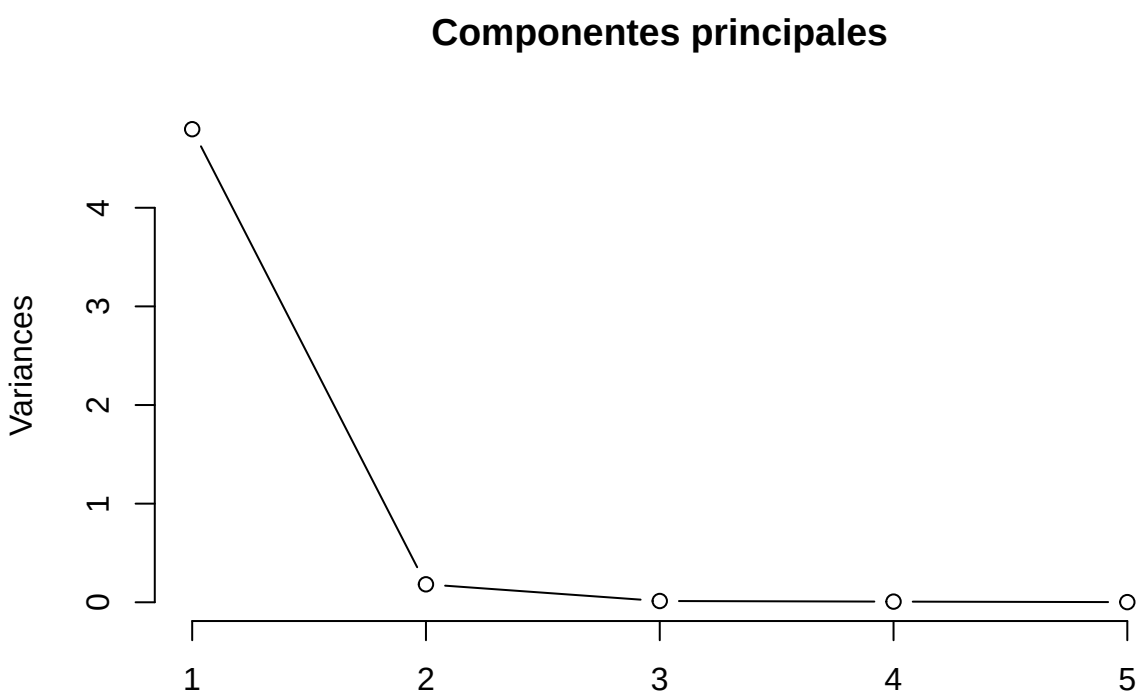
```
acp<-prcomp(x1, center=TRUE, scale=TRUE)
acp
```

```
## Standard deviations (1, .., p=5):
## [1] 2.19004754 0.42683643 0.11387349 0.08279062 0.04099989
##
## Rotation (n x k) = (5 x 5):
##                                    PC1         PC2         PC3         PC4
## Tamaño del lóbulo frontal 0.4540551 -0.08648822 -0.84960538  0.25366033
## Anchura trasera           0.4224434  0.88814833  0.15498291  0.06204351
## Longitud del caparazón    0.4530579 -0.26767350  0.15100960 -0.43438217
## Ancho del caparazón       0.4551379 -0.13943381  0.08476682 -0.54187613
## Profundidad corporal      0.4505148 -0.33559118  0.47345716  0.67043973
##                                    PC5
## Tamaño del lóbulo frontal  0.01344998
```

```
## Anchura trasera            0.06974959
## Longitud del caparazón     0.71526055
## Ancho del caparazón       -0.68745367
## Profundidad corporal      -0.10374913
```

3.- Generación del gráfico **screeplot**

```
plot(acp, type="l", main = "Componentes principales")
```



4.- Visualización del resumen de la matriz **ACP**

```
summary(acp)
```

```
## Importance of components:
##                           PC1     PC2     PC3     PC4     PC5
## Standard deviation     2.1900 0.42684 0.11387 0.08279 0.04100
## Proportion of Variance 0.9593 0.03644 0.00259 0.00137 0.00034
## Cumulative Proportion  0.9593 0.99570 0.99829 0.99966 1.00000
```

Árboles de decisión

Paquetería y librería a utilizar

```
install.packages("DMwR2")
library(DMwR2)
```

1.- Se utiliza una semilla

```
set.seed(1234)
data(crabs, package="MASS")
```

2.- Se deja la columna *sp* y se elimina la columna *sex* e *index* puesto que no nos interesan

```
crabs <- crabs[,-cbind(2,3)]
```

3.- Se consiguen aquellas variables para elaborar el árbol

```
ct1<-rpartXse(sp ~., crabs)
ct1
```

```
## n= 200
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##   1) root 200 100 B (0.50000000 0.50000000)
##     2) FL< 17.45 135  47 B (0.65185185 0.34814815)
##       4) CW>=36.2 40   4 B (0.90000000 0.10000000)
##         8) FL< 16.65 29   0 B (1.00000000 0.00000000) *
##         9) FL>=16.65 11   4 B (0.63636364 0.36363636)
##          18) CL>=36.15 7   0 B (1.00000000 0.00000000) *
##          19) CL< 36.15 4   0 O (0.00000000 1.00000000) *
##       5) CW< 36.2 95  43 B (0.54736842 0.45263158)
##        10) BD< 12.15 62  13 B (0.79032258 0.20967742)
##          20) CW>=29.85 21   0 B (1.00000000 0.00000000) *
##          21) CW< 29.85 41  13 B (0.68292683 0.31707317)
##            42) FL< 12.25 34   6 B (0.82352941 0.17647059)
##              84) CW>=25.1 15   0 B (1.00000000 0.00000000) *
##              85) CW< 25.1 19   6 B (0.68421053 0.31578947)
##               170) BD< 8.95 14   1 B (0.92857143 0.07142857) *
##               171) BD>=8.95 5   0 O (0.00000000 1.00000000) *
##            43) FL>=12.25 7   0 O (0.00000000 1.00000000) *
##        11) BD>=12.15 33   3 O (0.09090909 0.90909091)
##          22) CW>=34.6 10   3 O (0.30000000 0.70000000)
##            44) FL< 15.15 3   0 B (1.00000000 0.00000000) *
##            45) FL>=15.15 7   0 O (0.00000000 1.00000000) *
##          23) CW< 34.6 23   0 O (0.00000000 1.00000000) *
##     3) FL>=17.45 65  12 O (0.18461538 0.81538462)
##       6) CW>=44.35 33  12 O (0.36363636 0.63636364)
##        12) FL< 19.85 11   0 B (1.00000000 0.00000000) *
##        13) FL>=19.85 22   1 O (0.04545455 0.95454545) *
##       7) CW< 44.35 32   0 O (0.00000000 1.00000000) *
```

El tamaño de n son 200 cangrejos y aquellas variables marcadas en asterisco son aquellas variables significativas que servirán para la elaboración del árbol.

Elaboración del árbol:

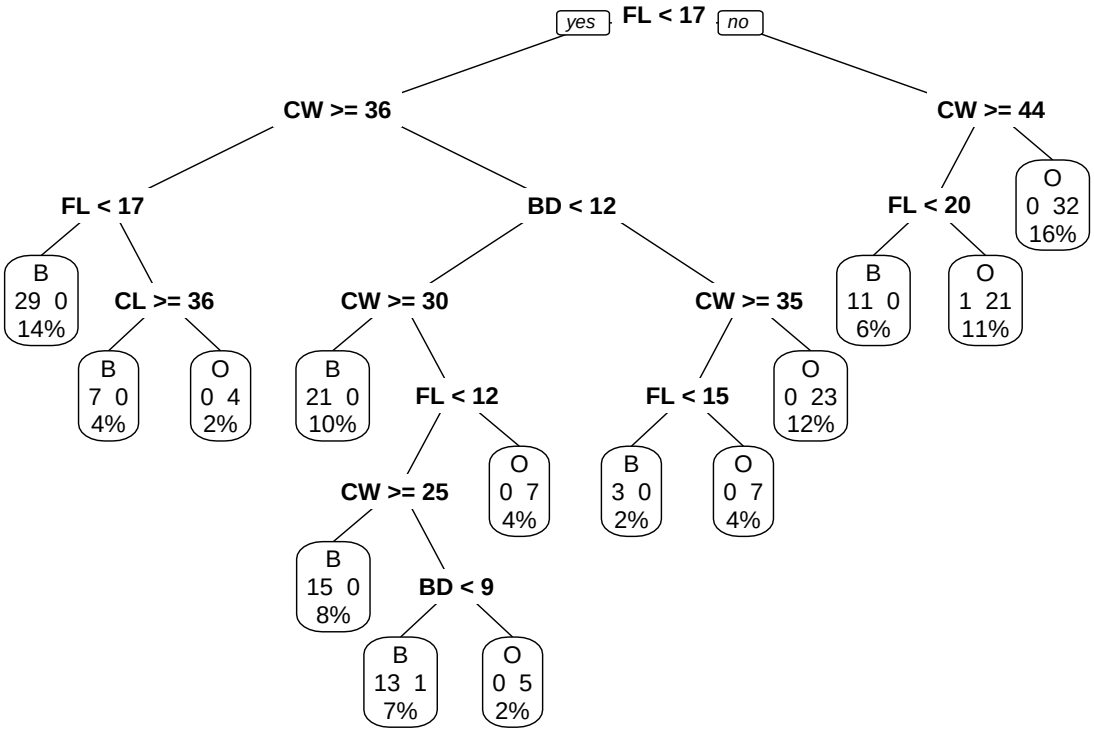
Paquetería y librería a utilizar

```
install.packages("rpart.plot")
library(rpart.plot)
library(rpart)
```

1.- Árbol de decisión

```
prp(ct1, type=0, extra=101)
```

```
## Warning: Cannot retrieve the data used to build the model (so cannot determine roundint and is.binary
## To silence this warning:
##     Call prp with roundint=FALSE,
##     or rebuild the rpart model with model=TRUE.
```



Para comprender el árbol, algunos puntos a considerar son:

-Los cuadritos son los nodos.

-Las líneas son los arcos.

-Las palabras en negritas son las condiciones.

-Si se cambia la semilla, el árbol cambia.

-Si no se pone una semilla, igual el árbol cambia.

-Lo que se hace es tomar una muestra de datos para esa semilla.

El árbol anterior presenta aquellas decisiones que se tomarán para clasificar a los cangrejos, por ejemplo: Si el tamaño del lóbulo frontal *(FL)* es menor que 17 entonces, si el ancho del caparazón *(CW)* es mayor o igual que 36 mm, se clasifica a 29 cangrejos en la especie azul, de lo contrario, si la longitud del caparazón *(CL)* es mayor o igual que 36 mm, se clasifica a 7 cangrejos en la especie azul y si no, 4 cangrejos en la especie naranja. En dado caso de que el tamaño del lóbulo frontal *(FL)* no sea menor que 17 mm y si el ancho del caparazón *(CW)* es mayor o igual a 44 mm, entonces, si el tamaño del lóbulo frontal *(FL)* es menor que 20 mm, se clasificará a 11 cangrejos en la especie azul y si no, se clasificarán 21 cangrejos en la especie naranja y 1 en la especie azul.

```
set.seed(1234)
```

2.- Se utiliza un tamaño de muestra 100

```
rndSample<-sample(1:nrow(crabs), 100)
```

3.- Muestra de datos/entrenamiento

```
tr <- crabs[rndSample,]
```

4.- Muestra de prueba

```
ts <- crabs[-rndSample, ]
ct <- rpartXse(sp ~., tr, se=0.5)
```

5.- Probabilidad de predicción con respecto a la muestra de entrenamiento

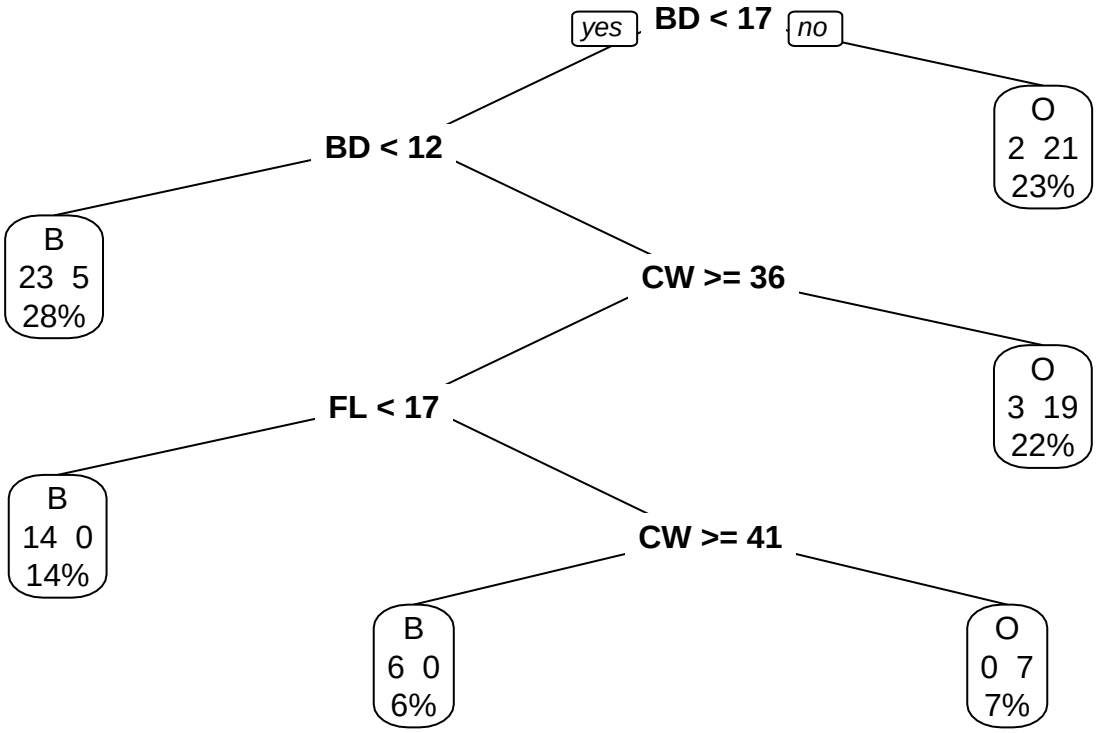
```
ps1<- predict(ct, ts)
head(ps1)
```

```
##            B         O
## 1  0.8214286 0.1785714
## 3  0.8214286 0.1785714
## 5  0.8214286 0.1785714
## 7  0.8214286 0.1785714
## 11 0.8214286 0.1785714
## 12 0.8214286 0.1785714
```

6.- Árbol de decisión

```
prp(ct, type=0, extra=101)
```

```
## Warning: Cannot retrieve the data used to build the model (so cannot determine roundint and is.binary
## To silence this warning:
##     Call prp with roundint=FALSE,
##     or rebuild the rpart model with model=TRUE.
```



El árbol anterior presenta un tamaño de muestra de 100 cangrejos, las decisiones a tomar para clasificarlos son: Si la profundidad corporal *(BD)* es menor que 17 mm y si es menor que 12 mm, se clasifica a 23 cangrejos en la especie azul y 5 en la especie naranja. Si la profundidad corporal *(BD)* no es menor que 17 mm, se clasifica a 21 cangrejos en la especie naranja y 2 en la especie azul.

```
ps2<-predict(ct, ts, type="class")
head(ps2)
```

```
##  1  3  5  7 11 12
##  B  B  B  B  B  B
## Levels: B O
```

7.- Matriz de confusión para ver con cuántos casos se equivoca

```
(cm<-table(ps2, ts$sp))
```

```
##
## ps2  B  O
##   B 44 10
##   O  8 38
```

8.- Número de error

```
100*(1-sum(diag(cm))/sum(cm))
```

```
## [1] 18
```

Se equivoca con 18 cangrejos.

9.- Evaluando con los datos que generaron el modelo

```
ps3<-predict(ct, tr, type="class")
head(ps3)
```

```
##  28  80 150 101 111 137
##   B   O   O   B   O   O
## Levels: B O
```

10.- Matriz de confusión para ver con cuántos casos se equivoca

```
(cm<-table(ps3, tr$sp))
```

```
##
## ps3  B  O
##   B 43  5
##   O  5 47
```

11.- Error

```
100*(1-sum(diag(cm))/sum(cm))
```

```
## [1] 10
```

CONCLUSIÓN

1.- Construcción del modelo de componentes principales:

z1 $= 0.454$*(Tamaño del lóbulo frontal)* $+ 0.422$*(Anchura trasera)* $+ 0.453$*(Longitud del caparazón)* $+ 0.455$*(Ancho del caparazón)* $+ 0.450$*(Profundidad corporal)*

2.- Interpretación del resultado:

Este componente conforma a la especie naranja de cangrejos. El modelo con un sólo componente principal, distingue entre el tamaño del lóbulo frontal, la anchura trasera, la longitud y el ancho del caparazón y la profundidad corporal que tienen los cangrejos de la especie naranja que de la especie azul. Como se explica más del 80% de la varianza, basta con que sea un sólo componente principal.

Los árboles de decisión mostraron aquellas decisiones a considerar para clasificar a cada cangrejo. Al considerar una muestra de 100 cangrejos, el árbol arrojó pocas decisiones, debido a que se trataba de la mitad de los datos. De igual manera, se puede ver el porcentaje que pertenece cada cangrejo en dicha clasificación.

REFERENCIAS

(s.a). (s.f) *Análisis de componentes principales (ACP): Principal component analysis (PCA). ¿Qué es el análisis de componentes principales?*. Compañia Telefónica Tech. Recuperado de: https://aiofthings.telefonicatech.com/recursos/datapedia/analisis-componentes-principales

López, J. F., (2019). *Árbol de decisión*. Economipedia.com. Recuperado de: https://economipedia.com/definiciones/arbol-de-decision.html

(s.a). (s.f). *Qué es un árbol de decisión y ejemplos*. edraw: A Wondershare Company. Recuperado de: https://www.edrawsoft.com/es/decision-tree/

Milborrowm, S. (2021). *rpart.plot: Plot 'rpart' Models: An Enhanced Version of 'plot.rpart'.* R package version 3.1.0. https://CRAN.R-project.org/package=rpart.plot

Torgo, L. (2016). *Data Mining with R, learning with case studies.* 2nd editionChapman and Hall/CRC. URL:http://ltorgo.github.io/DMwR2

Venables, W. N. & Ripley, B. D. (2002). *Modern Applied Statistics with S.* Fourth Edition. Springer, New York. ISBN 0-387-95457-0

Wickham et al., (2019). *Welcome to the tidyverse.* Journal of Open Source Software, 4(43), 1686, https://doi.org/10.21105/joss.01686