PCA

Arleth Michell Morales García

2022-03-25

Analisis de componentes principales

Introducción

El Analisis de componentes principales (**ACP**) es un método de reducción de la dimensionalidad de las variables originales

Matriz de trabajo

1.- Se trabajó con la matriz flores, extraída del paquete **datos** que se encuentra precargado en R.

```
install.packages("datos")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.1'
## (as 'lib' is unspecified)
```

```
library(datos)
```

2.- Se selecciona la matriz flores

```
flores<-datos::flores
```

Exploración de la matriz

1.- Dimesnión de la matriz La matriz cuenta con 150 obervaciones y 5 variables

```
dim(flores)
```

```
## [1] 150   5
```

2.- Tipo de variables

```
str(flores)
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Largo.Sepalo: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Ancho.Sepalo: num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Largo.Petalo: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Ancho.Petalo: num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Especie     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

3.- Nombre de las variables

```
colnames(flores)
```

```
## [1] "Largo.Sepalo" "Ancho.Sepalo" "Largo.Petalo" "Ancho.Petalo" "Especie"
```

4.- En busca de datos perdidos

```
anyNA(flores)
```

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

Tratamiento de la matriz

Generamos una nueva matriz ***Flores1***

```
flores1<-flores[51:100,1:4]
```

Desarrollar el PCA paso a paso

1.- Transformar la matriz en un data frame

```
flores1<-as.data.frame(flores1)
flores1
```

```
##     Largo.Sepalo Ancho.Sepalo Largo.Petalo Ancho.Petalo
## 51           7.0          3.2          4.7          1.4
## 52           6.4          3.2          4.5          1.5
## 53           6.9          3.1          4.9          1.5
## 54           5.5          2.3          4.0          1.3
## 55           6.5          2.8          4.6          1.5
## 56           5.7          2.8          4.5          1.3
## 57           6.3          3.3          4.7          1.6
## 58           4.9          2.4          3.3          1.0
## 59           6.6          2.9          4.6          1.3
## 60           5.2          2.7          3.9          1.4
## 61           5.0          2.0          3.5          1.0
## 62           5.9          3.0          4.2          1.5
## 63           6.0          2.2          4.0          1.0
## 64           6.1          2.9          4.7          1.4
## 65           5.6          2.9          3.6          1.3
## 66           6.7          3.1          4.4          1.4
## 67           5.6          3.0          4.5          1.5
## 68           5.8          2.7          4.1          1.0
## 69           6.2          2.2          4.5          1.5
## 70           5.6          2.5          3.9          1.1
## 71           5.9          3.2          4.8          1.8
## 72           6.1          2.8          4.0          1.3
## 73           6.3          2.5          4.9          1.5
## 74           6.1          2.8          4.7          1.2
## 75           6.4          2.9          4.3          1.3
## 76           6.6          3.0          4.4          1.4
## 77           6.8          2.8          4.8          1.4
## 78           6.7          3.0          5.0          1.7
## 79           6.0          2.9          4.5          1.5
## 80           5.7          2.6          3.5          1.0
## 81           5.5          2.4          3.8          1.1
## 82           5.5          2.4          3.7          1.0
## 83           5.8          2.7          3.9          1.2
## 84           6.0          2.7          5.1          1.6
## 85           5.4          3.0          4.5          1.5
## 86           6.0          3.4          4.5          1.6
```

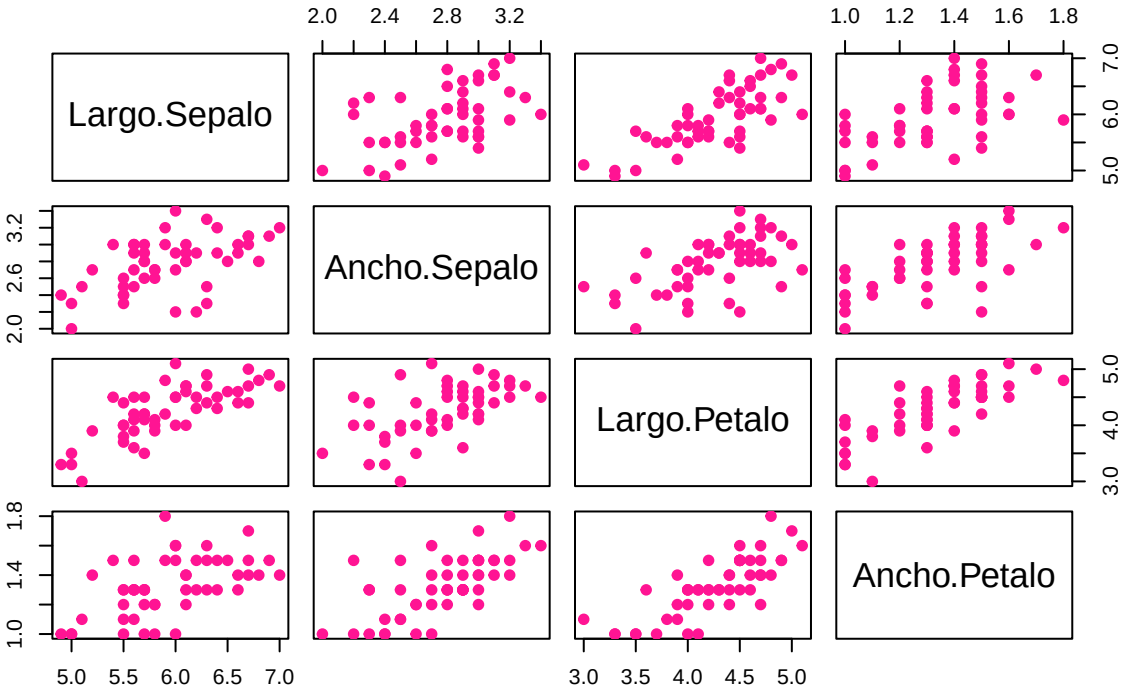
```
## 87          6.7         3.1          4.7         1.5
## 88          6.3         2.3          4.4         1.3
## 89          5.6         3.0          4.1         1.3
## 90          5.5         2.5          4.0         1.3
## 91          5.5         2.6          4.4         1.2
## 92          6.1         3.0          4.6         1.4
## 93          5.8         2.6          4.0         1.2
## 94          5.0         2.3          3.3         1.0
## 95          5.6         2.7          4.2         1.3
## 96          5.7         3.0          4.2         1.2
## 97          5.7         2.9          4.2         1.3
## 98          6.2         2.9          4.3         1.3
## 99          5.1         2.5          3.0         1.1
## 100         5.7         2.8          4.1         1.3
```

2.- Se definen n (numero de estados) y p (variables)

```
n<-dim(flores1)[1]
p<-dim(flores1)[2]
```

3. Generación del gráfico **scarteplot**

```
pairs(flores1 ,col="deeppink", pch=19,
      main="Variables originales")
```



Observando el grafico es posible apreciar que algunas variables como “ancho de sepalo” está ligeramente correlacionada con la variables “largo petalo”. La mayoria de las variables presenta una correlacion positiva

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

```
mu<-colMeans(flores1)
mu
```

```
## Largo.Sepalo Ancho.Sepalo Largo.Petalo Ancho.Petalo
##        5.936        2.770        4.260        1.326
```

```
s<-cov(flores1)
s
```

```
##              Largo.Sepalo Ancho.Sepalo Largo.Petalo Ancho.Petalo
## Largo.Sepalo   0.26643265   0.08518367   0.18289796   0.05577959
## Ancho.Sepalo   0.08518367   0.09846939   0.08265306   0.04120408
## Largo.Petalo   0.18289796   0.08265306   0.22081633   0.07310204
## Ancho.Petalo   0.05577959   0.04120408   0.07310204   0.03910612
```

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

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

```
## eigen() decomposition
## $values
## [1] 0.487873944 0.072384096 0.054776085 0.009790365
##
## $vectors
##           [,1]       [,2]        [,3]       [,4]
## [1,] 0.6867238  0.6690891 -0.26508336  0.1022796
## [2,] 0.3053470 -0.5674653 -0.72961786 -0.2289194
## [3,] 0.6236631 -0.3433270  0.62716496 -0.3159668
## [4,] 0.2149837 -0.3353051  0.06366081  0.9150409
```

5.1- Separación de la matriz de valores propios

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

```
## [1] 0.487873944 0.072384096 0.054776085 0.009790365
```

5.2 Separación de los vectores propios

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

```
##           [,1]       [,2]        [,3]       [,4]
## [1,] 0.6867238  0.6690891 -0.26508336  0.1022796
## [2,] 0.3053470 -0.5674653 -0.72961786 -0.2289194
## [3,] 0.6236631 -0.3433270  0.62716496 -0.3159668
## [4,] 0.2149837 -0.3353051  0.06366081  0.9150409
```

6.- Calcular la proporción de variabilidad

6.1.- Para la matriz de valores propios

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

```
## [1] 0.78081758 0.11584709 0.08766635 0.01566898
```

6.2.- Acumulada

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

```
## [1] 0.7808176 0.8966647 0.9843310 1.0000000
```

7.- Obtención de la matriz de correlaciones

```
R<-cor(flores1)
R
```

```
##              Largo.Sepalo Ancho.Sepalo Largo.Petalo Ancho.Petalo
## Largo.Sepalo    1.0000000    0.5259107    0.7540490    0.5464611
## Ancho.Sepalo    0.5259107    1.0000000    0.5605221    0.6639987
## Largo.Petalo    0.7540490    0.5605221    1.0000000    0.7866681
## Ancho.Petalo    0.5464611    0.6639987    0.7866681    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] 2.9263407 0.5462747 0.3949976 0.1323871
##
## $vectors
##            [,1]       [,2]       [,3]       [,4]
## [1,] -0.4823284  0.6107980 -0.4906296  0.3918772
## [2,] -0.4648460 -0.6727830 -0.5399025 -0.1994658
## [3,] -0.5345136  0.3068495  0.3402185 -0.7102042
## [4,] -0.5153375 -0.2830765  0.5933290  0.5497778
```

9.- Separacion de la matriz de valores propios a partir de la matriz de correlaciones

9.1.- Separacion de la matriz de valores propios

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

```
## [1] 2.9263407 0.5462747 0.3949976 0.1323871
```

9.2 Separación de los vectores propios

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

```
##            [,1]       [,2]       [,3]       [,4]
## [1,] -0.4823284  0.6107980 -0.4906296  0.3918772
## [2,] -0.4648460 -0.6727830 -0.5399025 -0.1994658
## [3,] -0.5345136  0.3068495  0.3402185 -0.7102042
## [4,] -0.5153375 -0.2830765  0.5933290  0.5497778
```

10.- Calculo de la proporción de variabilidad

10.1- Para la matriz de valores propios

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

```
## [1] 0.73158517 0.13656866 0.09874939 0.03309677
```

10.2.- Acumulada En este punto se seleccionan en número de componentes, siguiendo el criterio del 80% de la varianza explicada.

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

```
## [1] 0.7315852 0.8681538 0.9669032 1.0000000
```

11.- Calcular la media de los valores propios

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

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

Obtención de los coeficientes

12.- Centar los datos con respecto a la media 12.1.- Construcción de la matriz de 1

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

12.2.- Contrucción de la matriz centrada

```
X.cen<-as.matrix(flores1-ones%*%mu)
X.cen
```

```
##     Largo.Sepalo Ancho.Sepalo Largo.Petalo Ancho.Petalo
## 51         1.064         0.43         0.44        0.074
## 52         0.464         0.43         0.24        0.174
## 53         0.964         0.33         0.64        0.174
## 54        -0.436        -0.47        -0.26       -0.026
## 55         0.564         0.03         0.34        0.174
## 56        -0.236         0.03         0.24       -0.026
## 57         0.364         0.53         0.44        0.274
## 58        -1.036        -0.37        -0.96       -0.326
## 59         0.664         0.13         0.34       -0.026
## 60        -0.736        -0.07        -0.36        0.074
## 61        -0.936        -0.77        -0.76       -0.326
## 62        -0.036         0.23        -0.06        0.174
## 63         0.064        -0.57        -0.26       -0.326
## 64         0.164         0.13         0.44        0.074
## 65        -0.336         0.13        -0.66       -0.026
## 66         0.764         0.33         0.14        0.074
## 67        -0.336         0.23         0.24        0.174
## 68        -0.136        -0.07        -0.16       -0.326
## 69         0.264        -0.57         0.24        0.174
## 70        -0.336        -0.27        -0.36       -0.226
## 71        -0.036         0.43         0.54        0.474
## 72         0.164         0.03        -0.26       -0.026
## 73         0.364        -0.27         0.64        0.174
## 74         0.164         0.03         0.44       -0.126
## 75         0.464         0.13         0.04       -0.026
## 76         0.664         0.23         0.14        0.074
## 77         0.864         0.03         0.54        0.074
## 78         0.764         0.23         0.74        0.374
## 79         0.064         0.13         0.24        0.174
## 80        -0.236        -0.17        -0.76       -0.326
## 81        -0.436        -0.37        -0.46       -0.226
## 82        -0.436        -0.37        -0.56       -0.326
## 83        -0.136        -0.07        -0.36       -0.126
## 84         0.064        -0.07         0.84        0.274
## 85        -0.536         0.23         0.24        0.174
## 86         0.064         0.63         0.24        0.274
## 87         0.764         0.33         0.44        0.174
```

```
## 88         0.364        -0.47         0.14       -0.026
## 89        -0.336         0.23        -0.16       -0.026
## 90        -0.436        -0.27        -0.26       -0.026
## 91        -0.436        -0.17         0.14       -0.126
## 92         0.164         0.23         0.34        0.074
## 93        -0.136        -0.17        -0.26       -0.126
## 94        -0.936        -0.47        -0.96       -0.326
## 95        -0.336        -0.07        -0.06       -0.026
## 96        -0.236         0.23        -0.06       -0.126
## 97        -0.236         0.13        -0.06       -0.026
## 98         0.264         0.13         0.04       -0.026
## 99        -0.836        -0.27        -1.26       -0.226
## 100       -0.236         0.03        -0.16       -0.026
```

13.- Construcción de la matriz diagonal de covarianzas

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

```
##           [,1]       [,2]      [,3]       [,4]
## [1,] 0.2664327 0.00000000 0.0000000 0.00000000
## [2,] 0.0000000 0.09846939 0.0000000 0.00000000
## [3,] 0.0000000 0.00000000 0.2208163 0.00000000
## [4,] 0.0000000 0.00000000 0.0000000 0.03910612
```

14.- Construccion de la matriz centrada multiplicada por Dx^1/2

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

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

```
scores<-Y%*%eigen.vec
scores[1:10,]
```

```
##            [,1]       [,2]         [,3]        [,4]
## 51  2.498398022  0.1546668 -0.935158580 -0.05629969
## 52  1.543421364 -0.6465193 -0.861761442  0.42200924
## 53  2.642201627 -0.1098032 -0.352169754  0.32507678
## 54 -1.410740381  0.5188164  0.961337963  0.31099272
## 55  1.419955556  0.1333943  0.150394339  0.66638906
## 56  0.005474332 -0.4914314  0.363391127 -0.35033133
## 57  1.881838518 -1.2726644 -0.743796046  0.65748347
## 58 -3.366861014  0.5803346  0.006131683 -0.79833023
## 59  1.432877376  0.4212978 -0.197857358 -0.31218667
## 60 -1.444644905 -0.6899060  0.084085739  0.48970262
```

16.- Nombramos las columnas PC1..., PC4

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

17. Visualización de los scores

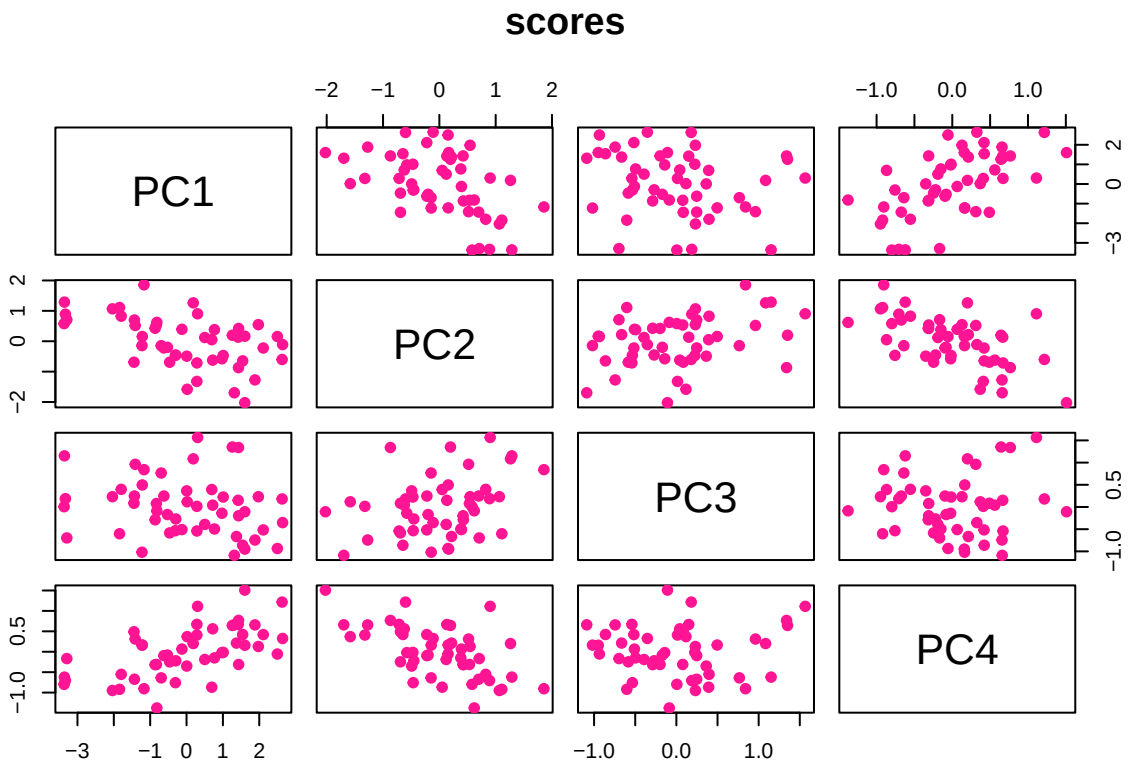
```
scores[1:10,]
```

```
##            PC1        PC2          PC3         PC4
## 51  2.498398022  0.1546668 -0.935158580 -0.05629969
## 52  1.543421364 -0.6465193 -0.861761442  0.42200924
## 53  2.642201627 -0.1098032 -0.352169754  0.32507678
## 54 -1.410740381  0.5188164  0.961337963  0.31099272
## 55  1.419955556  0.1333943  0.150394339  0.66638906
```

```
## 56  0.005474332 -0.4914314  0.363391127 -0.35033133
## 57  1.881838518 -1.2726644 -0.743796046  0.65748347
## 58 -3.366861014  0.5803346  0.006131683 -0.79833023
## 59  1.432877376  0.4212978 -0.197857358 -0.31218667
## 60 -1.444644905 -0.6899060  0.084085739  0.48970262
```

18. Generación del gráfico de los scores

```
pairs(scores, main="scores", col="deeppink", pch=19)
```



PCA VÍA SINTETIZADA

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

```
apply(flores1, 2, var)
```

```
## Largo.Sepalo Ancho.Sepalo Largo.Petalo Ancho.Petalo
##   0.26643265   0.09846939   0.22081633   0.03910612
```

2.- centrado por la media y escalada por la desviacion standar (dividir entre sd).

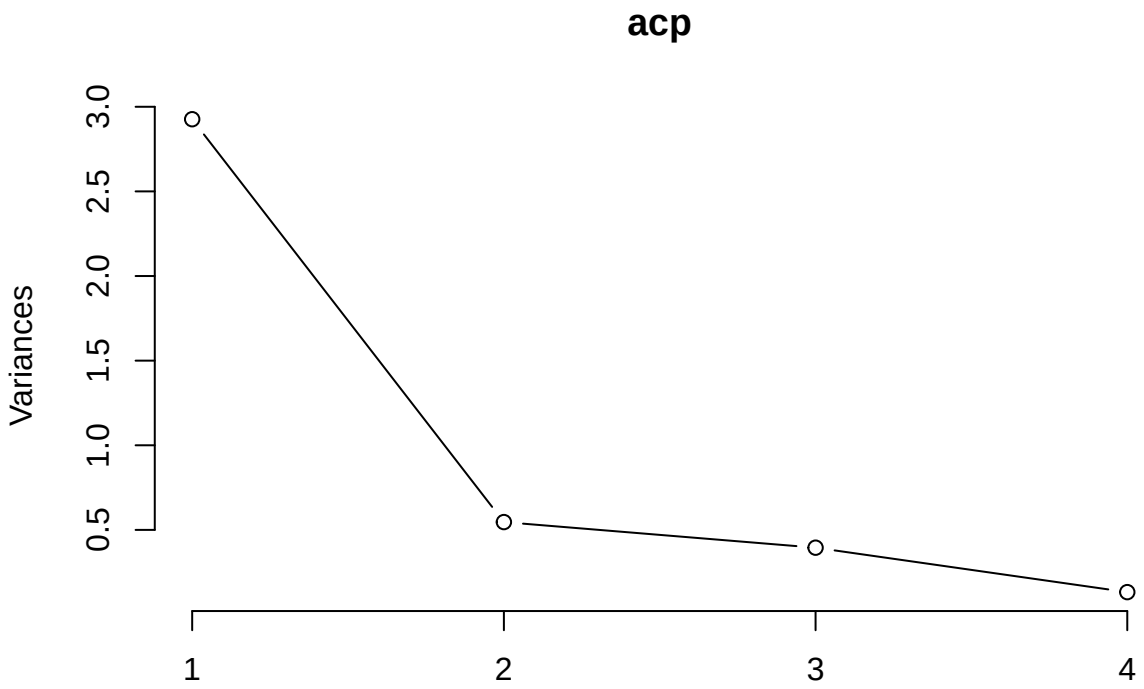
```
acp<-prcomp(flores1, center=TRUE, scale=TRUE)
acp
```

```
## Standard deviations (1, .., p=4):
## [1] 1.7106550 0.7391040 0.6284883 0.3638504
##
## Rotation (n x k) = (4 x 4):
##                     PC1        PC2        PC3        PC4
## Largo.Sepalo -0.4823284 -0.6107980  0.4906296  0.3918772
## Ancho.Sepalo -0.4648460  0.6727830  0.5399025 -0.1994658
## Largo.Petalo -0.5345136 -0.3068495 -0.3402185 -0.7102042
```

```
## Ancho.Petalo -0.5153375  0.2830765 -0.5933290  0.5497778
```

3.- Generación del gráfico screeplot

```
plot(acp, type="l")
```



Podemos observar que los componentes 1 y 2 son los más significativos.

4.- Resumen de la matriz ACP

```
summary(acp)
```

```
## Importance of components:
##                           PC1    PC2     PC3    PC4
## Standard deviation     1.7107 0.7391 0.62849 0.3639
## Proportion of Variance 0.7316 0.1366 0.09875 0.0331
## Cumulative Proportion  0.7316 0.8681 0.96690 1.0000
```

Construcción de los CP con las variables originales

Combinación lienal de las variables originales

$x1 = -0.4823284(var1) - 0.4648460(var2) - 0.5345136(var3) - 0.5153375(var4)$

El primer componente se distingue entre flores grandes y otras pequeñas. Todas presentan un signo negativo.

$x2 = -0.6107980(var1) + 0.6727830(var2) - 0.3068495(var3) + 0.2830765(var4)$

El segundo componente se distingue por presentar las flores por especies.