Análisis de Componentes Principales

Karina Itzel Rodríguez Conde

2022-03-24

ANÁLISIS DE COMPONENTES PRINCIPALES

Introducción

El Análisis de componentes principales **(ACP)** es un método de reducción de la dimensionalidad de las variables originales. Este análisis consiste en generar nuevas variables que sean resultado de la combinación lineal de las originales, consiguiendo de esta manera agrupar la mayor variación posible reduciendo su número. Es una manera de estudiar las relaciones que se presentan entre x variables correlacionadas y que se puede transformar el conjunto original de variables en otro conjunto de nuevas variables incorrelacionadas entre sí. De esta manera, en vez de tener muchas variables, tenemos sólo unas pocas que agrupan la mayor parte de la variación observada.

Matriz de trabajo

1.- Para esta práctica, se trabajó con la matriz flores, la cual fue extraída del paquete **datos** que se encuentra precargada en R.

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

```
library(datos)
```

2.- Se selecciona la matriz flores y se guarda en una variable, en este caso será x.

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

Exploración de la matriz

1.- Dimensión de la matriz:

```
dim(x)
```

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

La matriz cuenta con 150 observaciones y 5 variables.

2.- Tipo de variables:

```
str(x)
```

```
## '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(x)
```

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

4.- Saber si hay datos perdidos:

```
anyNA(x)
```

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

Para esta matriz no hay datos perdidos.

Tratamiento de la matriz

Se genera una nueva matriz *x1* que filtrará las variables cuantitativas de la especie Versicolor.

```
x1 <- x[51:100,1:4]
```

ANÁLISIS DE COMPONENTES PRINCIPALES (PCA) PASO A PASO

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

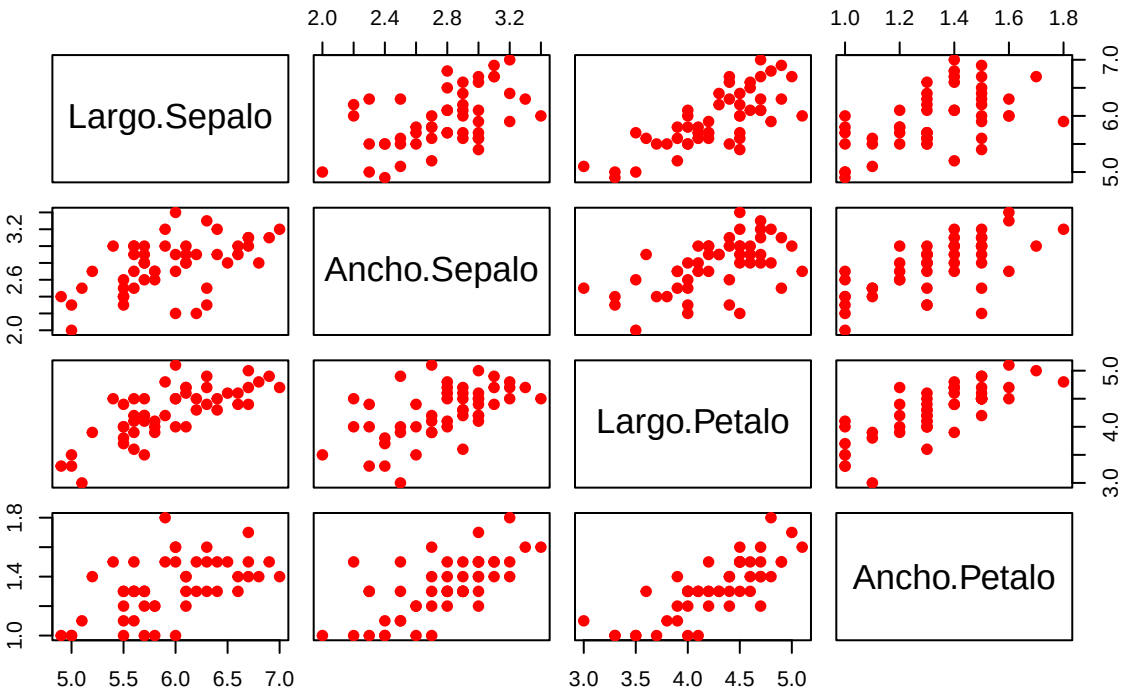
```
x1 <- as.data.frame(x1)
```

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 (Especie).

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



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

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

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

```
s <-cov(x1)
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** desde 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 la matriz de 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.- Proporción de variabilidad para cada valor:

6.1.- Para la matriz de valores propios.

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

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

6.2.- Proporción de variabilidad 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(x1)
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.- 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] 2.9263407 0.5462747 0.3949976 0.1323871
```

9.2.- Separación de la matriz de 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.- 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] 0.78081758 0.11584709 0.08766635 0.01566898
```

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] 0.7808176 0.8966647 0.9843310 1.0000000
```

En este caso, se seleccionan dos factores (0.896% de varianza explicada)

11.- Cálculo de la media de los valores propios.

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

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

Obtención de coeficientes

12.- 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 covrianzas.

```
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.- Construcción 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.R matriz de autovectores.

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

```
##            [,1]         [,2]        [,3]         [,4]
## 51  -2.32455278  0.518527321 -1.21059316  0.075191200
## 52  -1.79699308 -0.465213092 -0.48504815  0.199955742
## 53  -2.57106666  0.602046937 -0.49865033  0.038577169
## 54   1.46714905  0.359189046  0.95682822  0.288414020
## 55  -1.41164332  0.576018057  0.18051660  0.378999671
## 56  -0.02915352 -0.149647585  0.26845808 -0.633250224
## 57  -2.33977751 -0.810493078 -0.11721324  0.036211804
## 58   3.45770058 -0.592861742 -0.05182738 -0.006758222
## 59  -1.13202813  0.766244156 -0.68666085 -0.164670936
## 60   1.00808930 -1.061853727  0.78140281  0.235542894
## 61   3.72930250  0.513668902  0.68613800  0.021149805
## 62  -0.69226152 -0.823974223  0.11711656  0.400893274
## 63   1.92985776  1.594691271 -0.24648411 -0.102459451
## 64  -1.03915545  0.096734677  0.16103432 -0.417394083
## 65   0.93988525 -1.070075290 -0.46014981  0.587487920
## 66  -1.55484349  0.182031288 -0.97058589  0.364403810
## 67  -0.75317453 -0.983073087  0.61947364 -0.280274478
## 68   1.26232055  0.351326861 -0.84424984 -0.723260153
## 69  -0.12875333  1.442120020  1.42559495  0.683765246
## 70   1.71237679  0.269715849 -0.15480382 -0.167685639
## 71  -2.45281003 -1.290417031  1.10752004  0.200986462
## 72   0.16581167 -0.002815261 -0.47375199  0.426109347
## 73  -1.12159400  1.178451046  1.10398385 -0.035553205
## 74  -0.36982621  0.597427928 -0.26698347 -0.909854798
## 75  -0.60389762  0.333680772 -0.71375927  0.136896737
## 76  -1.31326471  0.278098640 -0.70348019  0.352048748
## 77  -1.65887251  1.204761121 -0.25987322  0.026475081
## 78  -2.87098618  0.358787884  0.53597948  0.355191858
## 79  -0.97881325 -0.295343487  0.41132067  0.086970539
## 80   2.18638635  0.055597243 -1.01154791  0.171200194
## 81   2.06770341  0.300483705  0.03990124 -0.028904800
## 82   2.44204823  0.378330936 -0.33253526 -0.155781733
## 83   0.96862226 -0.065565579 -0.38897934  0.135037320
## 84  -1.62562671  0.382106359  1.48986825 -0.414702129
## 85  -0.56628753 -1.219737987  0.80957711 -0.432114510
## 86  -1.98008661 -1.510489218 -0.14891326  0.047158606
## 87  -2.15668398  0.234783048 -0.45334812  0.189008940
## 88   0.26460970  1.567046623  0.48601687  0.291230540
## 89   0.22301078 -0.957977617 -0.27020063 -0.231756541
## 90   1.17087850 -0.069610556  0.61272029  0.161284112
## 91   0.82834885  0.120334348  0.43023297 -0.784837282
## 92  -1.07354289 -0.182964619 -0.08342027 -0.329823135
## 93   1.00300970  0.214133717 -0.14452475  0.047466372
```

```
## 94   3.51239235 -0.260129491  0.02517485  0.132726747
## 95   0.55366877 -0.249478719  0.31836189 -0.192197582
## 96   0.27641642 -0.631198947 -0.59288761 -0.584985261
## 97   0.16395472 -0.559945871 -0.12079777 -0.243407473
## 98  -0.41701062  0.097015872 -0.52365580 -0.014943295
## 99   3.20332484 -0.909641854 -0.33115082  0.812937395
## 100  0.42583783 -0.410845565 -0.02114443 -0.028706618
```

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

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

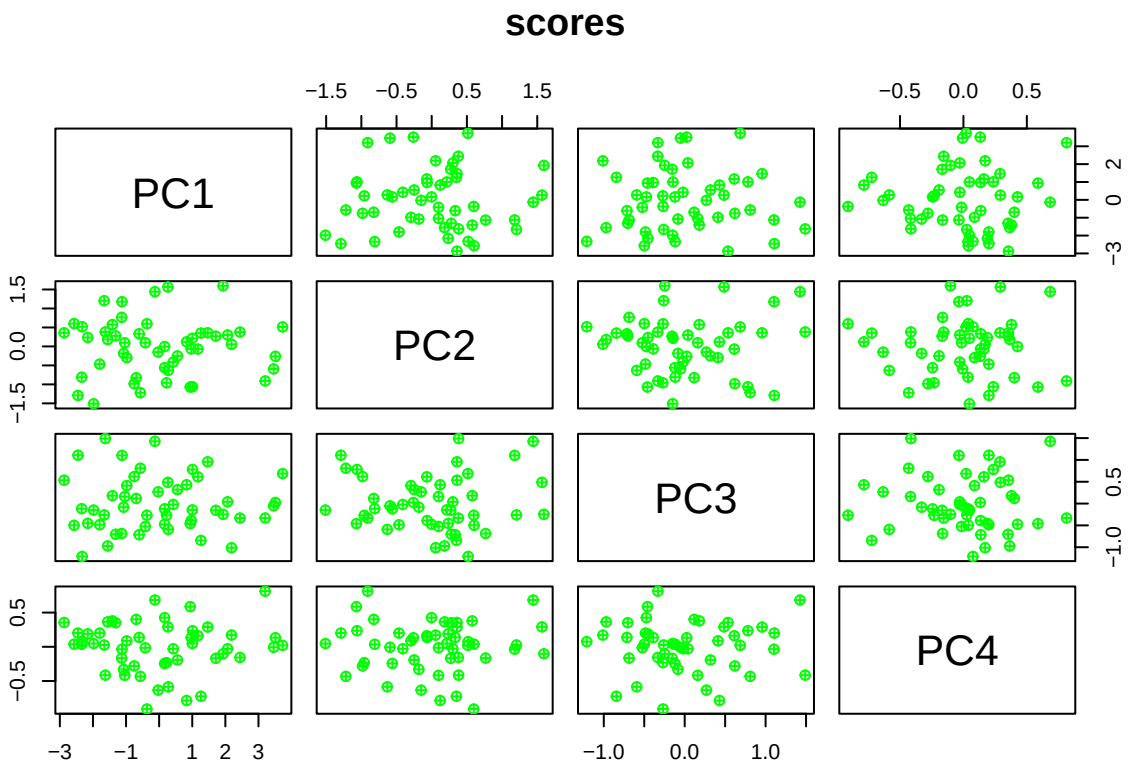
17.- Visualización de los scores.

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

```
##             PC1        PC2         PC3          PC4
## 51 -2.32455278  0.5185273 -1.21059316  0.075191200
## 52 -1.79699308 -0.4652131 -0.48504815  0.199955742
## 53 -2.57106666  0.6020469 -0.49865033  0.038577169
## 54  1.46714905  0.3591890  0.95682822  0.288414020
## 55 -1.41164332  0.5760181  0.18051660  0.378999671
## 56 -0.02915352 -0.1496476  0.26845808 -0.633250224
## 57 -2.33977751 -0.8104931 -0.11721324  0.036211804
## 58  3.45770058 -0.5928617 -0.05182738 -0.006758222
## 59 -1.13202813  0.7662442 -0.68666085 -0.164670936
## 60  1.00808930 -1.0618537  0.78140281  0.235542894
```

18.- Generacion del gráfico de los scores.

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



ANÁLISIS DE COMPONENTES PRINCIPALES VÍA SINTETIZADA

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

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

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

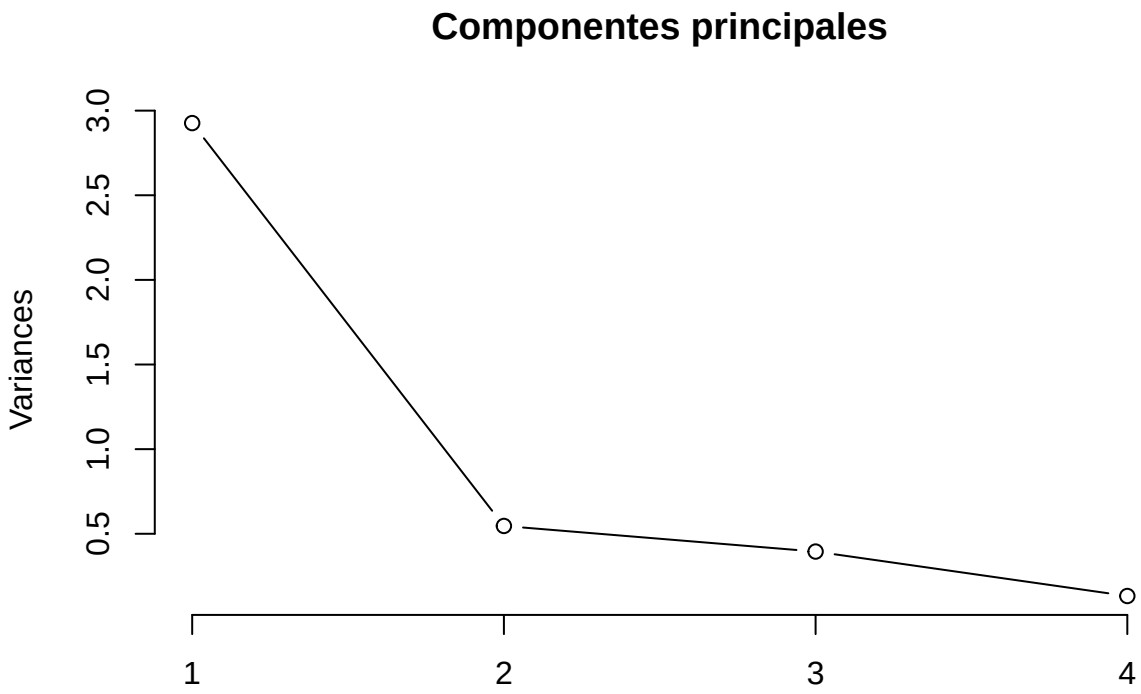
2.- Aplicación de la función **prcomp** para reducir la dimensionalidad y centrado por la media y escalada por la desviacion standar (dividir entre sd).

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

```
## Standard deviations (1, .., p=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", main = "Componentes principales")
```



4.- Visualización del 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 Componentes Principales con las variables originales

Combinación lineal de las variables originales.

1.- Elaboración del primer componente principal:

z1 = -0.482(Largo.Sepalo) - 0.464(Ancho.Sepalo) - 0.534(Largo.Petalo) - 0.515(Ancho.Petalo)

Este componente distingue entre flores grandes y pequeñas.

- Sépalo corto
- Sépalo angosto
- Pétalo corto
- Pétalo angosto

2.- Elaboración del segundo componente principal:

z2 = -0.610(Largo.Sepalo) + 0.672(Ancho.Sepalo) - 0.306(Largo.Petalo) + 0.283(Ancho.Petalo)

Este componente distingue flores por especie.

- Sépalo corto
- Sépalo ancho
- Pétalo corto
- Pétalo ancho