PCA

Edgar Ortiz Mota

2022-03-28

Análisis de componetes principales

Introducción

El análisis de componentes principales (*ACP*) es un método que sirve para reducir las dimenciones de las variables originales, para mejorar nuestro modelo.

Matriz de trabajo

1.- Se trabajo con la matriz *diamantes*, extraida del paquete *datos* que ya se encuentra precargado en R.

2.- Se selecciona la matriz (diamantes)

```
x= datos::diamantes
```

Exploración de la matriz

1.- Dimención de la matriz, la matriz cuenta con 53940 observaciones y 10 variables.

```
dim(x)
```

```
## [1] 53940    10
```

2.- Tipo de variables.

```
str(x)
```

```
## tibble [53,940 x 10] (S3: tbl_df/tbl/data.frame)
##  $ precio     : int [1:53940] 326 326 327 334 335 336 336 337 337 338 ...
##  $ quilate    : num [1:53940] 0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 0.23 ...
##  $ corte      : Ord.factor w/ 5 levels "Regular"<"Bueno"<..: 5 4 2 4 2 3 3 3 1 3 ...
##  $ color      : Ord.factor w/ 7 levels "D"<"E"<"F"<"G"<..: 2 2 2 6 7 7 6 5 2 5 ...
##  $ claridad   : Ord.factor w/ 8 levels "I1"<"SI2"<"SI1"<..: 2 3 5 4 2 6 7 3 4 5 ...
##  $ profundidad: num [1:53940] 61.5 59.8 56.9 62.4 63.3 62.8 62.3 61.9 65.1 59.4 ...
##  $ tabla      : num [1:53940] 55 61 65 58 58 57 57 55 61 61 ...
##  $ x          : num [1:53940] 3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ...
##  $ y          : num [1:53940] 3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 ...
##  $ z          : num [1:53940] 2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39 ...
```

3.-Nombre de las variables.

```
colnames(x)
```

```
##  [1] "precio"      "quilate"     "corte"       "color"       "claridad"
##  [6] "profundidad" "tabla"       "x"           "y"           "z"
```

4.- En busca de datos perdidos.

```
anyNA(x)
```

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

Tratamiento de la matriz

Se creara una nueva matriz donde solo tenga datos cuantitativos.

1.- Se seleccionaron los datos cuantitativos, para ello se seleccionaron manualmente.

```
x=diamantes [,c(1,2,6,7,8,9,10)]
```

PCA paso a paso

1.1.- Primero se transforma la matriz en un data frame

```
x= as.data.frame(x)
```

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

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

3.- Se genera el grafico *scaterplot*

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

Grafico

```
#pairs(x, main = "Datos diamantes", pch = 21, bg = "green3",
#lower.panel=NULL, labels=c("LS","AS","LP","AP"), font.labels=2, cex.labels=4.5)
```

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

```
mu= colMeans(x)
mu
```

```
##       precio      quilate  profundidad        tabla            x            y
## 3932.7997219    0.7979397   61.7494049   57.4571839    5.7311572    5.7345260
##            z
##    3.5387338
```

```
s = cov(x)
s
```

```
##                    precio      quilate  profundidad        tabla            x
## precio       1.591563e+07 1.742765e+03 -60.85371214 1133.3180641 3958.0214908
## quilate      1.742765e+03 2.246867e-01   0.01916653    0.1923645    0.5184841
## profundidad -6.085371e+01 1.916653e-02   2.05240384   -0.9468399   -0.0406413
## tabla        1.133318e+03 1.923645e-01  -0.94683994    4.9929481    0.4896429
## x            3.958021e+03 5.184841e-01  -0.04064130    0.4896429    1.2583472
## y            3.943271e+03 5.152478e-01  -0.04800857    0.4689723    1.2487893
## z            2.424713e+03 3.189168e-01   0.09596797    0.2379960    0.7684875
##                         y            z
## precio      3943.27081043 2.424713e+03
## quilate        0.51524782 3.189168e-01
## profundidad   -0.04800857 9.596797e-02
## tabla          0.46897228 2.379960e-01
## x              1.24878933 7.684875e-01
```

```
## y              1.30447161 7.673196e-01
## z              0.76731958 4.980109e-01
```

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

```
es = eigen(s)
es
```

```
## eigen() decomposition
## $values
## [1] 1.591563e+07 5.213080e+00 1.782627e+00 6.728544e-01 3.796773e-02
## [6] 1.579564e-02 6.076678e-03
##
## $vectors
##               [,1]          [,2]          [,3]          [,4]          [,5]
## [1,]  9.999999e-01 -9.340588e-05  4.969435e-05  0.0003884624 -1.965194e-05
## [2,]  1.095002e-04  1.268053e-02 -3.095226e-02 -0.1881334278  1.693347e-01
## [3,] -3.823523e-06 -2.855246e-01 -9.546313e-01  0.0616683380 -3.426591e-02
## [4,]  7.120789e-05  9.562977e-01 -2.805972e-01  0.0817901805 -6.533849e-03
## [5,]  2.486877e-04  4.450580e-02 -3.885860e-02 -0.6063073272  4.761949e-01
## [6,]  2.477609e-04  4.173598e-02 -3.277768e-02 -0.6626667125 -7.420649e-01
## [7,]  1.523479e-04  9.328059e-03 -8.001227e-02 -0.3838975099  4.389613e-01
##               [,6]          [,7]
## [1,]  1.540157e-05 -0.0000282091
## [2,] -2.727369e-01  0.9275922119
## [3,] -4.147808e-02 -0.0213840864
## [4,]  3.927639e-03 -0.0034998183
## [5,] -5.189888e-01 -0.3644031583
## [6,]  8.266921e-02  0.0237073198
## [7,]  8.047951e-01  0.0758383874
```

5.1.- Separación de la matriz de valores propios.

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

```
## [1] 1.591563e+07 5.213080e+00 1.782627e+00 6.728544e-01 3.796773e-02
## [6] 1.579564e-02 6.076678e-03
```

5.2.- Separación de la matriz de vectores propios.

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

```
##               [,1]          [,2]          [,3]          [,4]          [,5]
## [1,]  9.999999e-01 -9.340588e-05  4.969435e-05  0.0003884624 -1.965194e-05
## [2,]  1.095002e-04  1.268053e-02 -3.095226e-02 -0.1881334278  1.693347e-01
## [3,] -3.823523e-06 -2.855246e-01 -9.546313e-01  0.0616683380 -3.426591e-02
## [4,]  7.120789e-05  9.562977e-01 -2.805972e-01  0.0817901805 -6.533849e-03
## [5,]  2.486877e-04  4.450580e-02 -3.885860e-02 -0.6063073272  4.761949e-01
## [6,]  2.477609e-04  4.173598e-02 -3.277768e-02 -0.6626667125 -7.420649e-01
## [7,]  1.523479e-04  9.328059e-03 -8.001227e-02 -0.3838975099  4.389613e-01
##               [,6]          [,7]
## [1,]  1.540157e-05 -0.0000282091
## [2,] -2.727369e-01  0.9275922119
## [3,] -4.147808e-02 -0.0213840864
## [4,]  3.927639e-03 -0.0034998183
## [5,] -5.189888e-01 -0.3644031583
```

```
## [6,]  8.266921e-02  0.0237073198
## [7,]  8.047951e-01  0.0758383874
```

6.- Calcular la proporcion de la variabilidad.

6.1.- Para la matriz de valores propios.

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

```
## [1] 9.999995e-01 3.275445e-07 1.120047e-07 4.227630e-08 2.385561e-09
## [6] 9.924600e-10 3.818054e-10
```

6.2.- Acumulada

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

```
## [1] 0.9999995 0.9999998 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
```

7.- Obtención de la matriz de correlaciones.

```
R<-cor(x)
R
```

```
##                precio    quilate profundidad      tabla           x
## precio       1.0000000 0.92159130 -0.01064740  0.1271339  0.88443516
## quilate      0.9215913 1.00000000  0.02822431  0.1816175  0.97509423
## profundidad -0.0106474 0.02822431  1.00000000 -0.2957785 -0.02528925
## tabla        0.1271339 0.18161755 -0.29577852  1.0000000  0.19534428
## x            0.8844352 0.97509423 -0.02528925  0.1953443  1.00000000
## y            0.8654209 0.95172220 -0.02934067  0.1837601  0.97470148
## z            0.8612494 0.95338738  0.09492388  0.1509287  0.97077180
##                      y          z
## precio       0.86542090 0.86124944
## quilate      0.95172220 0.95338738
## profundidad -0.02934067 0.09492388
## tabla        0.18376015 0.15092869
## x            0.97470148 0.97077180
## y            1.00000000 0.95200572
## z            0.95200572 1.00000000
```

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

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

```
## eigen() decomposition
## $values
## [1] 4.76391480 1.28586808 0.69081126 0.17375333 0.04030722 0.03294659 0.01239871
##
## $vectors
##               [,1]         [,2]         [,3]        [,4]        [,5]
## [1,] -0.4255192667  0.035257945  0.105449477  0.84977817 -0.05377206
## [2,] -0.4524454941  0.034696011  0.005494814  0.06835945  0.13399948
## [3,]  0.0009161301  0.730679714 -0.672829294  0.04724800 -0.08873829
## [4,] -0.0995160875 -0.675067376 -0.728069469  0.05954060 -0.01037614
## [5,] -0.4532125054 -0.003512550  0.039508824 -0.24299509  0.08898016
## [6,] -0.4472649035 -0.002157912  0.054188788 -0.32846061 -0.77405793
## [7,] -0.4459536619  0.089035176 -0.039603439 -0.31700727  0.60339656
```

```
##             [,6]         [,7]
## [1,]  0.27330947  0.082814286
## [2,] -0.76815114 -0.425880295
## [3,] -0.01445027  0.055600264
## [4,]  0.02526831  0.002049255
## [5,] -0.19846061  0.828658219
## [6,]  0.21526655 -0.208857094
## [7,]  0.49867040 -0.279957944
```

9.-Separación de la matriz de valores propios a partir de las correlaciones.

9.1.- Separación de la matriz de valores propios.

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

```
## [1] 4.76391480 1.28586808 0.69081126 0.17375333 0.04030722 0.03294659 0.01239871
```

9.2.- Separación de la matriz de vectores propios.

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

```
##               [,1]         [,2]         [,3]        [,4]        [,5]
## [1,] -0.4255192667  0.035257945  0.105449477  0.84977817 -0.05377206
## [2,] -0.4524454941  0.034696011  0.005494814  0.06835945  0.13399948
## [3,]  0.0009161301  0.730679714 -0.672829294  0.04724800 -0.08873829
## [4,] -0.0995160875 -0.675067376 -0.728069469  0.05954060 -0.01037614
## [5,] -0.4532125054 -0.003512550  0.039508824 -0.24299509  0.08898016
## [6,] -0.4472649035 -0.002157912  0.054188788 -0.32846061 -0.77405793
## [7,] -0.4459536619  0.089035176 -0.039603439 -0.31700727  0.60339656
##             [,6]         [,7]
## [1,]  0.27330947  0.082814286
## [2,] -0.76815114 -0.425880295
## [3,] -0.01445027  0.055600264
## [4,]  0.02526831  0.002049255
## [5,] -0.19846061  0.828658219
## [6,]  0.21526655 -0.208857094
## [7,]  0.49867040 -0.279957944
```

10.- Cálculo de la proporcion de la variablilidad.

10.1.- Para la la matriz de valores propios.

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

```
## [1] 0.680559258 0.183695440 0.098687323 0.024821905 0.005758174 0.004706656
## [7] 0.001771245
```

10.2.- Acumulada. En este punto se seleccionan en número de componentes, siguiendo el criterio del 80% de la varianza explicada. Para este ejemplo se van a seleccionar 2 factores (0.8642% de varianza explicada).

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

```
## [1] 0.6805593 0.8642547 0.9629420 0.9877639 0.9935221 0.9982288 1.0000000
```

11.- Calcular 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 diagonal de covarianzas

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

13.- Contrucción de la matriz diagonal de covarianzas.

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

```
##          [,1]      [,2]     [,3]     [,4]     [,5]     [,6]      [,7]
## [1,] 15915629 0.0000000 0.000000 0.000000 0.000000 0.000000 0.0000000
## [2,]        0 0.2246867 0.000000 0.000000 0.000000 0.000000 0.0000000
## [3,]        0 0.0000000 2.052404 0.000000 0.000000 0.000000 0.0000000
## [4,]        0 0.0000000 0.000000 4.992948 0.000000 0.000000 0.0000000
## [5,]        0 0.0000000 0.000000 0.000000 1.258347 0.000000 0.0000000
## [6,]        0 0.0000000 0.000000 0.000000 0.000000 1.304472 0.0000000
## [7,]        0 0.0000000 0.000000 0.000000 0.000000 0.000000 0.4980109
```

14.- Construción de la matriz centrada

```
Y = X.cen<-as.matrix(x)-ones%*%mu
```

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

Se muestrann las primeras 10 observaciones.

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

```
##           [,1]      [,2]      [,3]      [,4]     [,5]      [,6]      [,7]
##  [1,] 1537.350 -125.8002 -378.5030 -3063.816 194.4472 -985.9718 -299.2707
##  [2,] 1536.904 -131.1036 -381.7329 -3063.442 194.5637 -985.8584 -299.3313
##  [3,] 1535.894 -135.8880 -382.5696 -3062.604 194.5647 -985.4397 -299.3255
##  [4,] 1533.307 -126.8673 -380.4333 -3056.999 193.8635 -983.6647 -298.4786
##  [5,] 1532.703 -126.1638 -380.9260 -3056.217 193.7245 -983.3619 -298.2969
##  [6,] 1532.884 -125.8430 -379.7827 -3055.144 193.8195 -983.1920 -298.3885
##  [7,] 1532.875 -126.2093 -379.4444 -3055.173 193.8432 -983.1875 -298.4094
##  [8,] 1532.499 -125.1108 -377.6042 -3054.551 193.7946 -982.9402 -298.3060
##  [9,] 1532.179 -126.8266 -384.1501 -3053.876 193.6565 -982.8554 -298.1843
## [10,] 1531.609 -130.9658 -380.1857 -3053.383 193.8522 -982.5250 -298.3434
```

16.- Nombarmos las columnas.

```
colnames(scores)<-c("PC1","PC2","PC3","PC4","PC5",
                    "PC6", "PC7")
```

17.- Vizualización de los scores

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

```
##            PC1       PC2       PC3       PC4      PC5       PC6       PC7
##  [1,] 1537.350 -125.8002 -378.5030 -3063.816 194.4472 -985.9718 -299.2707
##  [2,] 1536.904 -131.1036 -381.7329 -3063.442 194.5637 -985.8584 -299.3313
##  [3,] 1535.894 -135.8880 -382.5696 -3062.604 194.5647 -985.4397 -299.3255
##  [4,] 1533.307 -126.8673 -380.4333 -3056.999 193.8635 -983.6647 -298.4786
##  [5,] 1532.703 -126.1638 -380.9260 -3056.217 193.7245 -983.3619 -298.2969
##  [6,] 1532.884 -125.8430 -379.7827 -3055.144 193.8195 -983.1920 -298.3885
##  [7,] 1532.875 -126.2093 -379.4444 -3055.173 193.8432 -983.1875 -298.4094
##  [8,] 1532.499 -125.1108 -377.6042 -3054.551 193.7946 -982.9402 -298.3060
##  [9,] 1532.179 -126.8266 -384.1501 -3053.876 193.6565 -982.8554 -298.1843
## [10,] 1531.609 -130.9658 -380.1857 -3053.383 193.8522 -982.5250 -298.3434
```

18.- Generación del grafico de los scores.

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

##Acp via sintetizada

```
apply(x, 2, var)
```

```
##       precio      quilate  profundidad        tabla            x            y
## 1.591563e+07 2.246867e-01 2.052404e+00 4.992948e+00 1.258347e+00 1.304472e+00
##            z
## 4.980109e-01
```

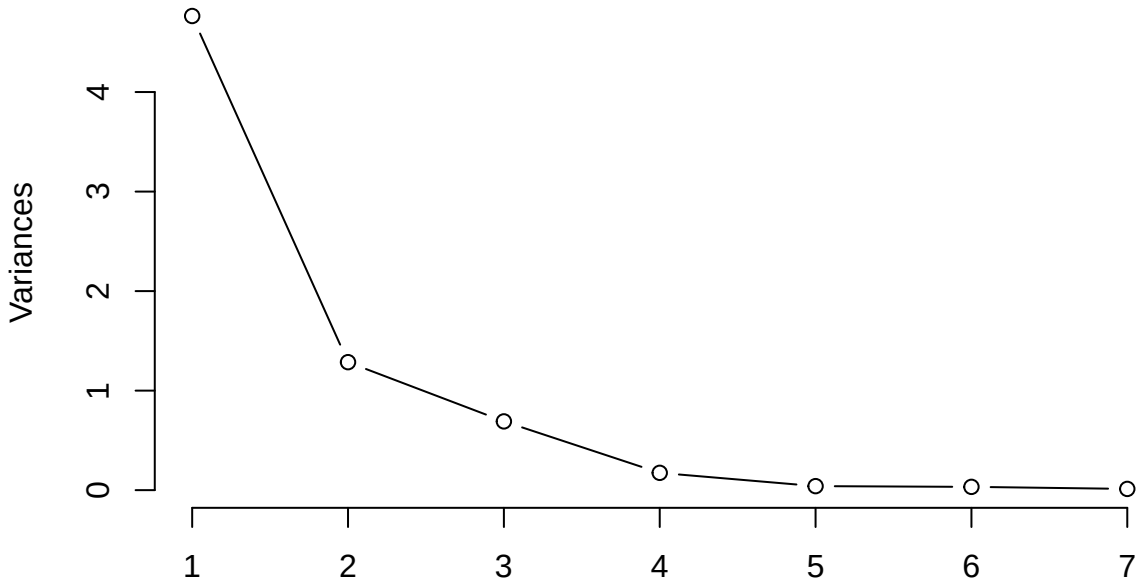
2.- Aplicar la funcion *prcomp* para reducir la dimensionalidad y centrado por la media y escala por la desciacion estandar (dividir entre sd).

```
acp= prcomp(x, center=TRUE, scale=TRUE)
acp
```

```
## Standard deviations (1, .., p=7):
## [1] 2.1826394 1.1339612 0.8311506 0.4168373 0.2007666 0.1815120 0.1113495
##
## Rotation (n x k) = (7 x 7):
##                         PC1          PC2          PC3         PC4         PC5
## precio       0.4255192667  0.035257945  0.105449477  0.84977817 -0.05377206
## quilate      0.4524454941  0.034696011  0.005494814  0.06835945  0.13399948
## profundidad -0.0009161301  0.730679714 -0.672829294  0.04724800 -0.08873829
## tabla        0.0995160875 -0.675067376 -0.728069469  0.05954060 -0.01037614
## x            0.4532125054 -0.003512550  0.039508824 -0.24299509  0.08898016
## y            0.4472649035 -0.002157912  0.054188788 -0.32846061 -0.77405793
## z            0.4459536619  0.089035176 -0.039603439 -0.31700727  0.60339656
##                        PC6          PC7
## precio       0.27330947 -0.082814286
## quilate     -0.76815114  0.425880295
## profundidad -0.01445027 -0.055600264
## tabla        0.02526831 -0.002049255
## x           -0.19846061 -0.828658219
## y            0.21526655  0.208857094
## z            0.49867040  0.279957944
```

3.- Generacion del grafico *screeplot*.

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



4.- Resumen de la matriz *acp*

```
summary(acp)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     2.1826 1.1340 0.83115 0.41684 0.20077 0.18151 0.11135
## Proportion of Variance 0.6806 0.1837 0.09869 0.02482 0.00576 0.00471 0.00177
## Cumulative Proportion  0.6806 0.8642 0.96294 0.98776 0.99352 0.99823 1.00000
```

Construcción de los CP con las variables originales

combinación lineal de las variables originales

z1= 0.4255192667(var1) + 0.4524454941(var2) -0.0009161301(var3) + 0.0995160875(var4) + 0.4532125054(var5) + 0.4472649035(var6) + 0.4459536619(var7)

El primer componente distingue entre

precio quilate profundidad tabla x y

z2= 0.035257945(var1) + 0.034696011(var2) + 0.730679714(var3) -0.675067376(var4) -0.003512550(var5) -0.002157912(var6) + 0.089035176(var7)

El segundo componente distingue

precio quilate profundidad tabla x y

Nota: los graficos en el punto 3 y 18 no se pudieron apreciar en este archivo, es por eso que se agrego como si fuera comentario con el uso del #