Componentes Principales

Andrea Piñeiro

2022-10-11

Parte A

De los siguientes datos:

```
x1 = c(2.5, 0.5, 2.2, 1.9, 3.1, 2.3, 2, 1, 1.5, 1.1)
x2 = c(2.4, 0.7, 2.9, 2.2, 3.0, 2.7, 1.6, 1.1, 1.6, 0.9)

M = cbind(x1, x2)
M
```

```
##        x1  x2
##  [1,] 2.5 2.4
##  [2,] 0.5 0.7
##  [3,] 2.2 2.9
##  [4,] 1.9 2.2
##  [5,] 3.1 3.0
##  [6,] 2.3 2.7
##  [7,] 2.0 1.6
##  [8,] 1.0 1.1
##  [9,] 1.5 1.6
## [10,] 1.1 0.9
```

1) Obtenga la matriz de datos centrados en sus medias

```
meanx1 = mean(x1)
meanx2 = mean(x2)

mx1 = c(rep(meanx1, 10))
mx2 = c(rep(meanx2, 10))

M1 = cbind(mx1, mx2)
M1 = M - M1
M1
```

```
##          x1    x2
##  [1,]  0.69  0.49
##  [2,] -1.31 -1.21
##  [3,]  0.39  0.99
##  [4,]  0.09  0.29
##  [5,]  1.29  1.09
```

```
##  [6,]  0.49  0.79
##  [7,]  0.19 -0.31
##  [8,] -0.81 -0.81
##  [9,] -0.31 -0.31
## [10,] -0.71 -1.01
```

2) Obtenga la matriz de varianza-covarianza de la matriz de datos centrados

```
mcov = cov(M1)
mcov
```

```
##           x1        x2
## x1 0.6165556 0.6154444
## x2 0.6154444 0.7165556
```

3) Obtenga los valores propios y vectores propios de la matriz de varianza-covarianza de la matriz de datos centrados.

```
L = eigen(mcov)
eigVal = L$values
eigVec = L$vectors
L
```

```
## eigen() decomposition
## $values
## [1] 1.2840277 0.0490834
##
## $vectors
##           [,1]       [,2]
## [1,] 0.6778734 -0.7351787
## [2,] 0.7351787  0.6778734
```

4) Obtenga las matrices transpuestas de los vectores propios y la traspuesta de la matriz de datos centrados.

```
t_v = t(eigVec)
t_M1 = t(M1)
t_v
```

```
##            [,1]      [,2]
## [1,]  0.6778734 0.7351787
## [2,] -0.7351787 0.6778734
```

```
t_M1
```

```
##    [,1]  [,2] [,3] [,4] [,5] [,6]  [,7]  [,8]  [,9] [,10]
## x1 0.69 -1.31 0.39 0.09 1.29 0.49  0.19 -0.81 -0.31 -0.71
## x2 0.49 -1.21 0.99 0.29 1.09 0.79 -0.31 -0.81 -0.31 -1.01
```

5) Multiplique la matriz transpuesta de los vectores propios con la transpuesta de la matriz de datos centrados.

```
prod = t_v %*% t_M1
rownames(prod) = c("CP1", "CP2")
t(prod)
```

```
##               CP1         CP2
##  [1,]  0.82797019 -0.17511531
##  [2,] -1.77758033  0.14285723
##  [3,]  0.99219749  0.38437499
##  [4,]  0.27421042  0.13041721
##  [5,]  1.67580142 -0.20949846
##  [6,]  0.91294910  0.17528244
##  [7,] -0.09910944 -0.34982470
##  [8,] -1.14457216  0.04641726
##  [9,] -0.43804614  0.01776463
## [10,] -1.22382056 -0.16267529
```

6) Interprete los resultados.

Al obtener la covarianza podemos ver que la covarianza entre ambas variables es positiva, por lo que se mueven en la msima dirección; y es una dependencia de 0.61.

Como se pudo observar de los vectores de Eigen tendríamos: Y1 = 0.6778 * x1 + 0.7351 * x2 Y2 = -0.7351 * x1 + 0.67787 * x2

Los tamaños relativos de los coeficientes son suficientemente grandes, y como podemos observar para Y2 tienen signos opuestos, por lo que ambas variables contribuyen en la interpretación de Y

Como podemos observar el primer componente principal esta correlacionado fuertemente en la misma dirección (mayor a 0.5) con 4 variables (1, 3, 5 y 6); y correlacionado fuertemente en dirección opuesta con 3 variables (2, 8 y 10), lo que significa que el componente principal aumenta cuando estas disminuyen. En este componente principal, la mayoría de las variables tienen influencia en el valor del mismo.

A diferencia, el segundo componente principal no tiene variables con relación fuerte ya sea positiva o negativa. La mayor es la variable 3 con un valor de 0.38437499, el cual aún es muy bajo, pero significa que el componente principal incrementa cuando esta variable lo hace.

Parte B

```
cpa <- prcomp(M, scale=TRUE)
names(cpa)
```

```
## [1] "sdev"     "rotation" "center"   "scale"    "x"
```

```
print("desviaciones estándar: ")
```

```
## [1] "desviaciones estándar: "
```

```
cpa$sdev
```

```
## [1] 1.3877785 0.2721594
```

```r
print("medias: ")
```

```
## [1] "medias: "
```

```r
print("center y scale dan las medias y desv estándar previa estandarización: ")
```

```
## [1] "center y scale dan las medias y desv estándar previa estandarización: "
```

```r
cpa$center
```

```
##   x1   x2
## 1.81 1.91
```

```r
cpa$scale
```

```
##        x1        x2
## 0.7852105 0.8464960
```

```r
print("Los coeficientes de la combinación lineal normalizada de componete")
```

```
## [1] "Los coeficientes de la combinación lineal normalizada de componete"
```

```r
cpa$rotation
```

```
##           PC1        PC2
## x1 -0.7071068  0.7071068
## x2 -0.7071068 -0.7071068
```

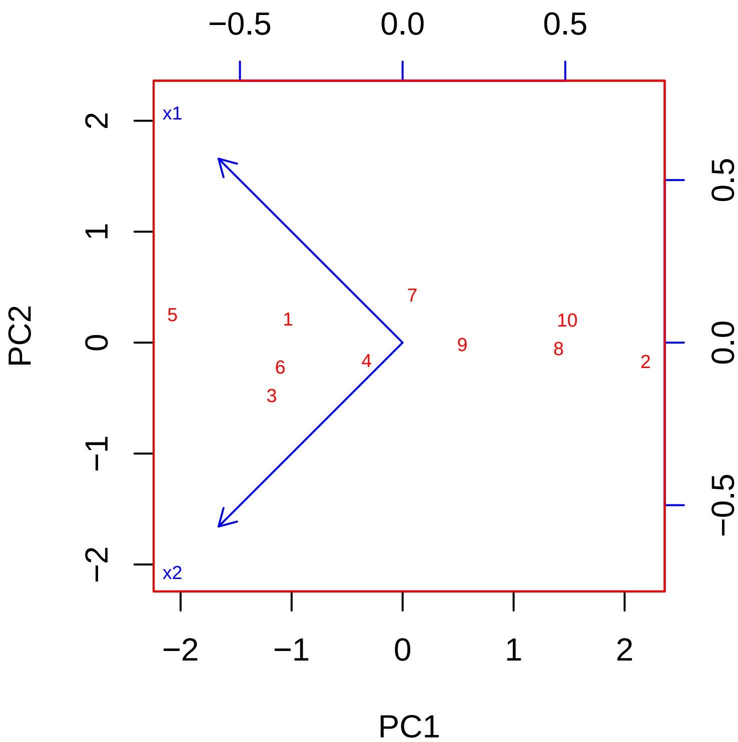
```r
print("Los datos por sustituidos en la combinación lineal de vectores propios:")
```

```
## [1] "Los datos por sustituidos en la combinación lineal de vectores propios:"
```

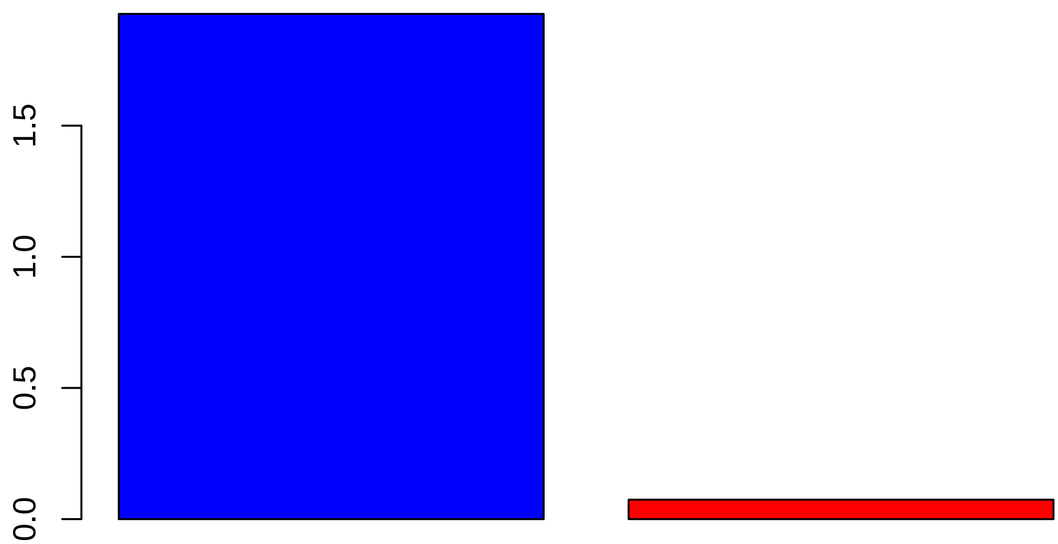
```r
cpa$x
```

```
##               PC1         PC2
##  [1,] -1.03068029  0.21205314
##  [2,]  2.19045016 -0.16894230
##  [3,] -1.17818776 -0.47577321
##  [4,] -0.32329464 -0.16119898
##  [5,] -2.07219947  0.25117173
##  [6,] -1.10117414 -0.21865330
##  [7,]  0.08785251  0.43005447
##  [8,]  1.40605089 -0.05281009
##  [9,]  0.53811824 -0.02021127
## [10,]  1.48306451  0.20430982
```

```r
biplot(x = cpa, scale = 0, cex =0.6, col = c("red", "blue"))
```




```r
barplot(cpa$sdev ^ 2, col = c("blue", "red"))
```




```r
summary(cpa)
```

```
## Importance of components:
##                          PC1     PC2
## Standard deviation     1.388 0.27216
## Proportion of Variance 0.963 0.03704
## Cumulative Proportion  0.963 1.00000
```

Esta es una gran alternativa para obtener los componentes principales usando funciones de R Con estos resultados de igual manera obtuvimos que el componente principal 1 tiene varias variables con correlación fuerte; y el componente principal 2 de igual manera no tiene ninguna variable con correlación alta (mayor a 0.5).