Componentes Principales

Felipe Yépez

2022-10-12

PARTE A

De los siguientes datos:

x1: 2.5, 0.5, 2.2, 1.9, 3.1, 2.3, 2, 1, 1.5, 1.1

x2: 2.4, 0.7, 2.9, 2.2, 3.0, 2.7, 1.6, 1.1, 1.6, 0.9

1. Obtenga una matriz de datos centrados en sus medias.

```
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)

m1 = c(rep(mean(x1), 10))
m2 = c(rep(mean(x2), 10))
M1 = cbind(m1, m2)
M_centrada = M - M1
M_centrada
```

```
##          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(M_centrada)
mcov
```

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

Se puede observar covarianza positiva entre ambas variables por lo que se puede saber que se mueven en la misma dirección.

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

```
L = eigen(mcov)
valores_propios = L$values
valores_propios
```

```
## [1] 1.2840277 0.0490834
```

Los eigen values miden la distancia de cada eje o variable, en este caso de las covarianzas. Se obtiene un valor mayor para las covarianzas de x1.

```
vectores_propios = L$vectors
vectores_propios
```

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

El eigen vector determina la dirección del eigen value en el espacio.

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

```
t_vectores_propios = t(vectores_propios)
t_vectores_propios
```

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

```
t_M_centrada = t(M_centrada)
t_M_centrada
```

```
##    [,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.

```
t_CP = t_vectores_propios %*% t_M_centrada
rownames(t_CP) = c("CP1", "CP2")
CP = t(t_CP)
CP
```

```
##              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. Lo que se obtuvo al multiplicar la matriz de datos centrada por sus eigen vectors, fue realizar una transformación en el orden de los eigen values obtenidos para sus covarianzas. Es decir se dio una transformación.

El primer componente principal muestra correlación fuerte positiva (>0.5) en las variables 1,3,5 y 6. LOs valores negativos obtenidos muestran lo contrario, es decir que el componente principal aumenta cuando dichas variables disminuyen. Se obtienen valores de alta correlación que muestran que la mayor parte de las variables tienen influencia en el primer componente.

El segundo componente muestra que no existen variables fuertes o influyentes en el componente principal 2.

PARTE B

Aplique a los mismos datos las fórmulas de R para Componentes principales e interpreta resultados.

```
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
```

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

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

```
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 componente")
```

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

```r
cpa$rotation
```

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

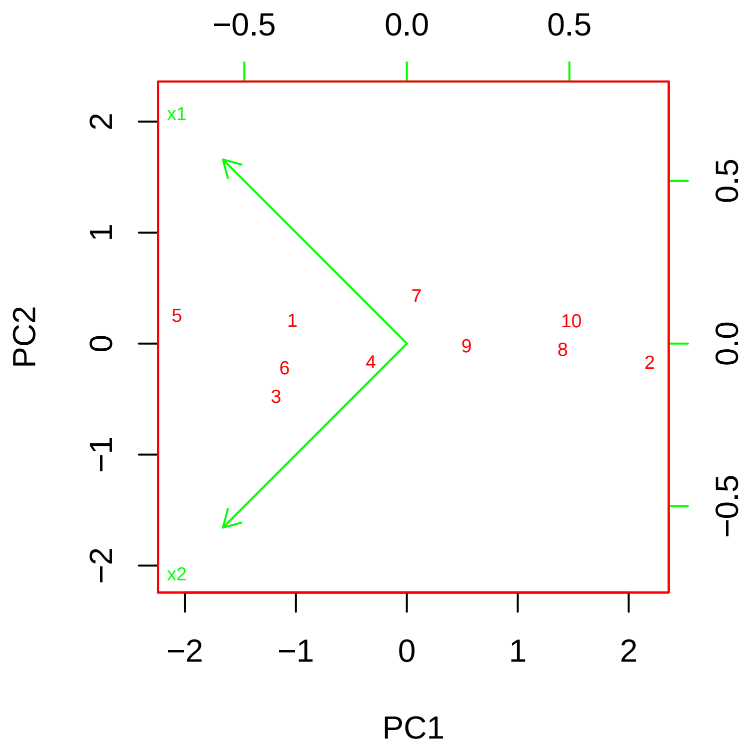
```r
print("Los datos por sustituidos en la combinación lineal de vectores propios:")
```

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

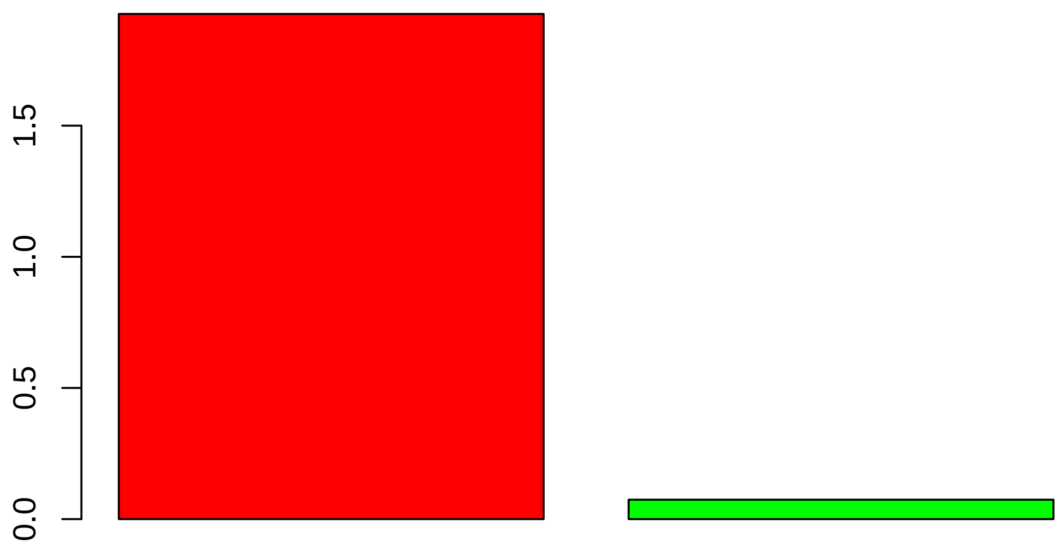
```r
cpa$x
```

```
##               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", "green"))
```




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




```
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
```

De igual forma se obtienen 2 componentes principales al intentar reducir la dimensionalidad. Estas funciones de R permiten realizar el proceso de una forma más sencilla y automatizada. En la gráfica de barras se muestra la desviación estandar elevada al cuadrado, obteniendo mayor valor en x1.