# 1-Práctica de Análisis BIPLOT en R

Consideremos los datos de la matriz X= $\begin{pmatrix} -2 & -2 & -1 \\ 0 & 1 & 0 \\ 2 & 2 & 2 \\ 0 & -1 & -1 \end{pmatrix}$ recogida en la siguiente representación gráfica Biplot. Vamos a calcular las coordenadas de las filas y columnas F y G, en cada uno de los 3 tipos de Biplot (SQ, JK, GH), así como la representación gráfica Biplot en R, por último calcularemos la estimación de la matriz X obtenida a través de esta técnica.

Pasos a realizar en la práctica:

1. Crea el fichero de texto biplot1.txt con los datos de la matriz X

```
> X <- read.table("biplot1.txt")
> X
  V1 V2 V3
1 -2 -2 -1
2  0  1  0
3  2  2  2
4  0 -1 -1
```

2. Realiza el análisis de Componentes principales sobre X. *Como esta matriz está centrada, las desviaciones típicas de la función prcom, es decir los valores singulares de X'X, coinciden con las raíces cuadradas de los autovalores de la DVS de la matriz de covarianzas, es decir 2.7253=√7.4273. Si realizamos un ACP no centrado, este resultado no se daría.*

```
> X_pca <- prcomp(X)
> X_pca
Standard deviations:
[1] 2.7253091 0.5773503 0.4892411 valores singulares de X'X

Rotation:
         PC1        PC2         PC3
V1 0.5735637  0.8164966 -0.06601555
V2 0.6544159 -0.4082483  0.63645361
V3 0.4927115 -0.4082483 -0.76848471

> summary(X_pca)
Importance of components:
                          PC1     PC2     PC3
Standard deviation     2.7253 0.57735 0.48924
Proportion of Variance 0.9284 0.04167 0.02992
Cumulative Proportion  0.9284 0.97008 1.00000
```

3. Comprobemos estos resultados:

   En el Análisis de Componentes Principales X'X= matriz de covarianzas siendo $x_{ij}=\dfrac{r_{ij}-\overline{r}_j}{\sqrt{n-1}}$ ya que la función prcomp, trabaja con n-1, como la matriz de partida X está centrada, la única transformación que hay que hacer es dividir por raíz de n-1

```
> cov(X)=X'X
         V1       V2       V3
V1 2.666667 2.666667 2.000000
V2 2.666667 3.333333 2.333333
V3 2.000000 2.333333 2.000000
```

   Veamos si a través de la anterior transformación llegamos a la matriz de covarianzas

```
> n <- 4
> X1 <- 1/sqrt(n-1)*(X)
```

```
         V1         V2         V3
1 -1.154701 -1.1547005 -0.5773503
2  0.000000  0.5773503  0.0000000
3  1.154701  1.1547005  1.1547005
4  0.000000 -0.5773503 -0.5773503
> X1 <- as.matrix(X1)
> (t(X1)%*%X1)
         V1       V2       V3
V1 2.666667 2.666667 2.000000
V2 2.666667 3.333333 2.333333
V3 2.000000 2.333333 2.000000
```

Efectivamente hemos llegado a la matriz de covarianzas

```
svd(cov(X))
$d
[1] 7.4273098 0.3333333 0.2393569 autovalores de X'X= cuadrado de los valores singulares

$u
           [,1]       [,2]        [,3]
[1,] -0.5735637  0.8164966  0.06601555
[2,] -0.6544159 -0.4082483 -0.63645361
[3,] -0.4927115 -0.4082483  0.76848471

$v
           [,1]       [,2]        [,3]
[1,] -0.5735637  0.8164966  0.06601555
[2,] -0.6544159 -0.4082483 -0.63645361
[3,] -0.4927115 -0.4082483  0.76848471
```

En el Análisis de Componentes Principales normalizado X'X=matriz de correlaciones donde $x_{ij}=\dfrac{r_{ij}-\overline{r}_j}{s_j\sqrt{n-1}}$

```
X_pca <- prcomp(X, scale=TRUE)
> X_pca
Standard deviations:
[1] 1.6661917 0.3668835 0.2986663 valores singulares de X'X

Rotation:
         PC1         PC2        PC3
V1 0.5739511  0.75162701 -0.3250185
V2 0.5820758 -0.09528683  0.8075322
V3 0.5759930 -0.65266934 -0.4921939

> summary(X_pca)
Importance of components:
                          PC1     PC2     PC3
Standard deviation     1.6662 0.36688 0.29867
Proportion of Variance 0.9254 0.04487 0.02973
Cumulative Proportion  0.9254 0.97027 1.00000

> cor(X)
          V1        V2        V3
V1 1.0000000 0.8944272 0.8660254
V2 0.8944272 1.0000000 0.9036961
V3 0.8660254 0.9036961 1.0000000

> svd(cor(X))
$d
[1] 2.77619493 0.13460353 0.08920154 autovalores de X'X= cuadrado de los valores singulares

$u
           [,1]        [,2]       [,3]
[1,] -0.5739511  0.75162701  0.3250185
[2,] -0.5820758 -0.09528683 -0.8075322
[3,] -0.5759930 -0.65266934  0.4921939

$v
           [,1]        [,2]       [,3]
[1,] -0.5739511  0.75162701  0.3250185
```

```
[2,] -0.5820758 -0.09528683 -0.8075322
[3,] -0.5759930 -0.65266934  0.4921939
```

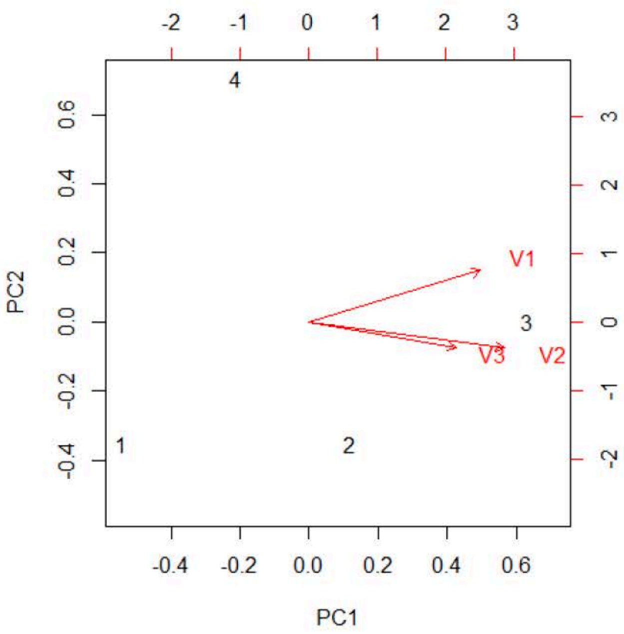
Comprobemos los resultados, realizando la transformación paso a paso

```
X <- as.matrix(X)
> X
     V1 V2 V3
[1,] -2 -2 -1
[2,]  0  1  0
[3,]  2  2  2
[4,]  0 -1 -1
> n <- 4
> TX <- matrix(c((1/sqrt(cov(X)[1,1]))/sqrt(n-1),0,0,0,(1/sqrt(cov(X)[2,2]))/sqrt(n-1),0,0,0,(1/sqrt(cov(X)[3,3]))/sqrt(n-1)),ncol=3)
> TX
          [,1]      [,2]      [,3]
[1,] 0.3535534 0.0000000 0.0000000
[2,] 0.0000000 0.3162278 0.0000000
[3,] 0.0000000 0.0000000 0.4082483
> X1N <- X%*%TX
> X1N
           [,1]       [,2]       [,3]
[1,] -0.7071068 -0.6324555 -0.4082483
[2,]  0.0000000  0.3162278  0.0000000
[3,]  0.7071068  0.6324555  0.8164966
[4,]  0.0000000 -0.3162278 -0.4082483
> t(X1N)%*%X1N
          [,1]      [,2]      [,3]
[1,] 1.0000000 0.8944272 0.8660254
[2,] 0.8944272 1.0000000 0.9036961
[3,] 0.8660254 0.9036961 1.0000000
```

Efectivamente obtenemos la matriz de correlaciones

4. Realiza la representación gráfica Biplot. Observa la posición de las 3 variables (columnas), representadas por vectores y de los 4 observaciones (filas) representadas por puntos

```
> biplot(X_pca)
```




5. Vamos a realizar los cálculos para cada uno de los 3 tipos BIPLOT, comenzamos calculando la (SVD) de la matriz X, (descomposición en valores singulares de X)

```
> svd(X)
```

```
$dX
[1] 4.7203739 1.0000000 0.8473905

$u
           [,1]          [,2]       [,3]
[1,] -0.6246689 -4.082483e-01 -0.4394567
[2,]  0.1386365 -4.082483e-01  0.7510747
[3,]  0.7290486  1.273997e-14 -0.4674271
[4,] -0.2430162  8.164966e-01  0.1558090

$v
          [,1]       [,2]        [,3]
[1,] 0.5735637  0.8164966 -0.06601555
[2,] 0.6544159 -0.4082483  0.63645361
[3,] 0.4927115 -0.4082483 -0.76848471
```

6. Vamos a comparar estos resultados de la DVS de X con los de la DVS de X'X, (XX1), y XX', (XX2)

```
> X <- as.matrix(X)
> XX1 <- t(X)%*%X
> XX1
   V1 V2 V3
V1  8  8  6
V2  8 10  7
V3  6  7  6
> XX2 <- X%*%t(X)
> XX2
     [,1] [,2] [,3] [,4]
[1,]    9   -2  -10    3
[2,]   -2    1    2   -1
[3,]  -10    2   12   -4
[4,]    3   -1   -4    2
> svd(XX1)
$d
[1] 22.2819293  1.0000000  0.7180707

$u
           [,1]       [,2]        [,3]
[1,] -0.5735637  0.8164966  0.06601555
[2,] -0.6544159 -0.4082483 -0.63645361
[3,] -0.4927115 -0.4082483  0.76848471

$v
           [,1]       [,2]        [,3]
[1,] -0.5735637  0.8164966  0.06601555
[2,] -0.6544159 -0.4082483 -0.63645361
[3,] -0.4927115 -0.4082483  0.76848471

> svd(XX2)
$d
[1] 2.228193e+01 1.000000e+00 7.180707e-01 6.222156e-16

$u
           [,1]          [,2]       [,3] [,4]
[1,] -0.6246689  4.082483e-01  0.4394567  0.5
[2,]  0.1386365  4.082483e-01 -0.7510747  0.5
[3,]  0.7290486 -5.477660e-16  0.4674271  0.5
[4,] -0.2430162 -8.164966e-01 -0.1558090  0.5

$v
           [,1]          [,2]       [,3] [,4]
[1,] -0.6246689  4.082483e-01  0.4394567 -0.5
[2,]  0.1386365  4.082483e-01 -0.7510747 -0.5
[3,]  0.7290486 -5.595025e-16  0.4674271 -0.5
[4,] -0.2430162 -8.164966e-01 -0.1558090 -0.5
```

Se observa la relación existente entre las DVS de X'X y XX', (que coinciden con las autodescomposiciones de X'X y XX') y la DVS de X, en el 1º caso los vectores singulares a izquierda y derecha son iguales, vectores U, para X'X y vectores V para XX', (aquí los nombres u y v están intercambiados en relación a la teoría), y también se observa que los valores singulares son las raíces cuadradas de los autovalores, ya que en la DVS, se obtienen valores singulares y en las autodescomposiciones, autovalores.

También se puede comprobar que las componentes principales del análisis realizado en el apartado 2, coinciden con los vectores U de X'X.

7. Vamos a calcular las coordenadas de las filas y columnas F y G de los 3 tipos de Biplot

   a. SQ Biplot (c=0.5) (biplot simétrico) $F=VD_\lambda^{1/4}$ $G= UD_\lambda^{1/4}$

```
> F1 <- svd(X)$u[,1]*sqrt(svd(X)$d[1])
> F1
[1] -1.3571819  0.3012074  1.5839617 -0.5279872
> F2 <- svd(X)$u[,2]*sqrt(svd(X)$d[2])
> F2
[1] -4.082483e-01 -4.082483e-01  1.273997e-14  8.164966e-01
> F <- cbind(F1,F2)
> F
             F1            F2
[1,] -1.3571819 -4.082483e-01
[2,]  0.3012074 -4.082483e-01
[3,]  1.5839617  1.273997e-14
[4,] -0.5279872  8.164966e-01
```

```
> G1 <- svd(X)$v[,1]*sqrt(svd(X)$d[1])
> G1
[1] 1.246149 1.421812 1.070486
> G2 <- svd(X)$v[,2]*sqrt(svd(X)$d[2])
> G2
[1]  0.8164966 -0.4082483 -0.4082483
> G <- cbind(G1,G2)
> G
           G1         G2
[1,] 1.246149  0.8164966
[2,] 1.421812 -0.4082483
[3,] 1.070486 -0.4082483
F%*%t(G)
             [,1]       [,2]       [,3]
[1,] -2.024583624 -1.7629902 -1.2861771
[2,]  0.042015836  0.5949268  0.4891049
[3,]  1.973851681  2.2520950  1.6956083
[4,]  0.008716106 -1.0840317 -0.8985361
```

   b. JK Biplot (c=1) $F=VD_\lambda^{1/2}$ $G= U$

```
> F1 <- svd(X)$u[,1]*svd(X)$d[1]
> F1
[1] -2.9486705  0.6544159  3.4413820 -1.1471273
> F2 <- svd(X)$u[,2]*svd(X)$d[2]
> F2
[1] -4.082483e-01 -4.082483e-01  1.273997e-14  8.164966e-01
> F <- cbind(F1,F2)
> F
             F1            F2
[1,] -2.9486705 -4.082483e-01
[2,]  0.6544159 -4.082483e-01
[3,]  3.4413820  1.273997e-14
[4,] -1.1471273  8.164966e-01
```

```
> G1 <- svd(X)$v[,1]
> G1
[1] 0.5735637 0.6544159 0.4927115
> G2 <- svd(X)$v[,2]
> G2
[1]  0.8164966 -0.4082483 -0.4082483
> G <- cbind(G1,G2)
> G
            G1         G2
[1,] 0.5735637  0.8164966
[2,] 0.6544159 -0.4082483
[3,] 0.4927115 -0.4082483
F%*%t(G)
             [,1]       [,2]       [,3]
[1,] -2.024583624 -1.7629902 -1.2861771
[2,]  0.042015836  0.5949268  0.4891049
[3,]  1.973851681  2.2520950  1.6956083
```

```
[4,]  0.008716106 -1.0840317 -0.8985361
```

c. GH Biplot (c=0) F=V    G= $UD_{\lambda}^{1/2}$

```
> F1 <- svd(X)$u[,1]
> F1
[1] -0.6246689  0.1386365  0.7290486 -0.2430162
> F2 <- svd(X)$u[,2]
> F2
[1] -4.082483e-01 -4.082483e-01  1.273997e-14  8.164966e-01
> F <- cbind(F1,F2)
> F
            F1           F2
[1,] -0.6246689 -4.082483e-01
[2,]  0.1386365 -4.082483e-01
[3,]  0.7290486  1.273997e-14
[4,] -0.2430162  8.164966e-01

> G1 <- svd(X)$v[,1]*svd(X)$d[1]
> G1
[1] 2.707435 3.089088 2.325782
> G2 <- svd(X)$v[,2]*svd(X)$d[2]
> G2
[1]  0.8164966 -0.4082483 -0.4082483
> G <- cbind(G1,G2)
> G
           G1         G2
[1,] 2.707435  0.8164966
[2,] 3.089088 -0.4082483
[3,] 2.325782 -0.4082483
F%*%t(G)
             [,1]       [,2]       [,3]
[1,] -2.024583624 -1.7629902 -1.2861771
[2,]  0.042015836  0.5949268  0.4891049
[3,]  1.973851681  2.2520950  1.6956083
[4,]  0.008716106 -1.0840317 -0.8985361
```

8. Es el momento de responder a la siguiente pregunta; ¿Por qué la 2ª columna de las matrices F y G de las coordenadas de las filas y columnas en los 3 tipos de Biplot, son iguales?
9. Vamos a obtener la matriz estimada de X por el Análisis Biplot, no por el producto de F y G, sino como la reconstrucción de 2º grado de la matriz X

```
>((svd(X)$d[1]))*(svd(X)$u[,1])%*%(t(svd(X)$v[,1]))+((svd(X)$d[2]))*(svd(X)$u[,2])%
*%(t(svd(X)$v[,2]))
             [,1]       [,2]       [,3]
[1,] -2.024583624 -1.7629902 -1.2861771
[2,]  0.042015836  0.5949268  0.4891049
[3,]  1.973851681  2.2520950  1.6956083
[4,]  0.008716106 -1.0840317 -0.8985361
```

10. Por último comprobar todos los resultados obtenidos con los que aparecen en el documento Biplot del enunciado.

11. Comprobación de resultados: **otra forma de hacerlo**

```
> X <- matrix(c(-2, 0, 2, 0, -2,1,2,-1,-1,0,2,-1), nrow=4)
> X
     [,1] [,2] [,3]
[1,]   -2   -2   -1
[2,]    0    1    0
[3,]    2    2    2
[4,]    0   -1   -1

> meanX <- apply(X,2,mean)
> meanX
[1] 0 0 0
```

```
> sdX <- apply(X,2,sd)
> sdX
[1] 1.632993 1.825742 1.414214

> t(X)%*%X
     [,1] [,2] [,3]
[1,]    8    8    6
[2,]    8   10    7
[3,]    6    7    6

> X%*%t(X)
     [,1] [,2] [,3] [,4]
[1,]    9   -2  -10    3
[2,]   -2    1    2   -1
[3,]  -10    2   12   -4
[4,]    3   -1   -4    2
> svd(t(X)%*%X)
$d
[1] 22.2819293  1.0000000  0.7180707

$u
           [,1]       [,2]        [,3]
[1,] -0.5735637  0.8164966  0.06601555
[2,] -0.6544159 -0.4082483 -0.63645361
[3,] -0.4927115 -0.4082483  0.76848471

$v
           [,1]       [,2]        [,3]
[1,] -0.5735637  0.8164966  0.06601555
[2,] -0.6544159 -0.4082483 -0.63645361
[3,] -0.4927115 -0.4082483  0.76848471

> svd(X%*%t(X))
$d
[1] 2.228193e+01 1.000000e+00 7.180707e-01 5.416122e-16

$u
           [,1]          [,2]       [,3] [,4]
[1,] -0.6246689  4.082483e-01  0.4394567  0.5
[2,]  0.1386365  4.082483e-01 -0.7510747  0.5
[3,]  0.7290486 -6.938894e-16  0.4674271  0.5
[4,] -0.2430162 -8.164966e-01 -0.1558090  0.5

$v
           [,1]          [,2]       [,3] [,4]
[1,] -0.6246689  4.082483e-01  0.4394567 -0.5
[2,]  0.1386365  4.082483e-01 -0.7510747 -0.5
[3,]  0.7290486 -1.110223e-16  0.4674271 -0.5
[4,] -0.2430162 -8.164966e-01 -0.1558090 -0.5

> lambda <- diag(sqrt(svd(t(X)%*%X)$d))
> lambda2 <- lambda[-3,1:2]
> lambda
         [,1] [,2]      [,3]
[1,] 4.720374    0 0.0000000
[2,] 0.000000    1 0.0000000
[3,] 0.000000    0 0.8473905
> lambda2
         [,1] [,2]
[1,] 4.720374    0
[2,] 0.000000    1
> lambda2^(1/2)
         [,1] [,2]
[1,] 2.172642    0
[2,] 0.000000    1

> V <- svd(X%*%t(X))$v[,1:2]
> V
           [,1]          [,2]
[1,] -0.6246689  4.082483e-01
[2,]  0.1386365  4.082483e-01
[3,]  0.7290486 -1.110223e-16
[4,] -0.2430162 -8.164966e-01

> F <- V%*%lambda2^(1/2)
> round(F,2)
```

```
      [,1]  [,2]
[1,] -1.36  0.41
[2,]  0.30  0.41
[3,]  1.58  0.00
[4,] -0.53 -0.82

> U <- svd(t(X)%*%X)$u[,1:2]

> G <- U%*%lambda2^(1/2)
> round(G,2)
      [,1]  [,2]
[1,] -1.25  0.82
[2,] -1.42 -0.41
[3,] -1.07 -0.41

> F <- V
> round(F,2)
      [,1]  [,2]
[1,] -0.62  0.41
[2,]  0.14  0.41
[3,]  0.73  0.00
[4,] -0.24 -0.82

> G <- U%*%lambda2
> round(G,2)
      [,1]  [,2]
[1,] -2.71  0.82
[2,] -3.09 -0.41
[3,] -2.33 -0.41

> F <- V%*%lambda2
> round(F,2)
      [,1]  [,2]
[1,] -2.95  0.41
[2,]  0.65  0.41
[3,]  3.44  0.00
[4,] -1.15 -0.82

> G <- U
> round(G,2)
      [,1]  [,2]
[1,] -0.57  0.82
[2,] -0.65 -0.41
[3,] -0.49 -0.41

> X <- -F%*%t(G)
> round(X,1)
     [,1] [,2] [,3]
[1,]   -2 -1.8 -1.3
[2,]    0  0.6  0.5
[3,]    2  2.3  1.7
[4,]    0 -1.1 -0.9
```