Los datos suministrados con el software del curso y que aparecen en el fichero denominado `datos4`, corresponden a 47 pares de observaciones (x_i, y_i), dados en Rousseeuw y Leroy (1987) página 27, de 47 estrellas del cúmulo CYG OB1, en las que se observó x_i =*logaritmo de la temperatura superficial de la estrella* **así como** y_i =*logaritmo de su intensidad luminosa*. **Determinar un Análisis de Componentes Principales Clásico y un Análisis de Componentes Principales Robusto.**

La nube de puntos resultante de su representación (figura 1) obtenida mediante la secuencia de instrucciones que aparece a continuación, se conoce como diagrama de Hertzsprung-Russell. En dicha figura pueden apreciarse claramente cuatro outliers, fuera de la *secuencia principal*, correspondientes a las estrellas números 11, 20, 30 y 34, conocidas como *gigantes rojas*. La estrella número 7 puede calificarse de dudosa.

```
> datos4<-matrix(scan("a:\\datos4"),ncol=2, byrow=T)
> plot(datos4[,1],datos4[,2],pch=16)
> text(datos4[,1],datos4[,2],1:47,adj=-1,cex=0.8,col=2)
```

Si realizáramos un análisis de componentes principales clásico ejecutaríamos `(1)` o equivalentemente `(2)` ó `(3)`, después de abrir el módulo `mva`. Los resultados de las componentes principales (la posible diferencia de signos es irrelevante) aparecen en `(4)`

```
> library(mva)
> M<-cov.wt(datos4)
> prcomp(datos4)                                          (1)
> robuscp(datos4)$loadings                                (2)
> robuscp(datos4,covmat=M)$loadings                       (3)

Desviaciones típicas:                                     (4)
[1] 0.5755684 0.2821791

Rotaciones:
            PC1        PC2
[1,]  0.1402946 -0.9901098
[2,] -0.9901098 -0.1402946
```

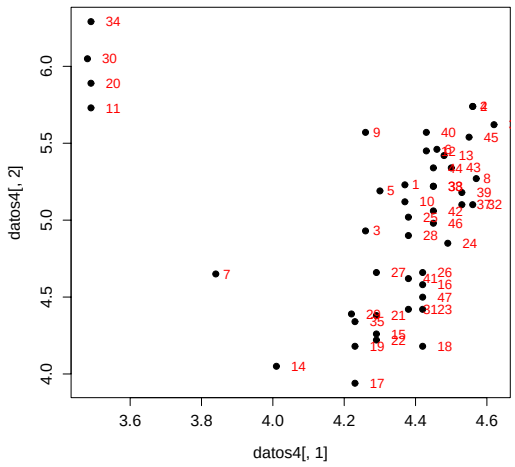

Figure 1: : Diagrama de Hertzsprung-Russell para el ejemplo

Como la medias de cada una de las dos variables son

```
> mean(datos4[,1])
[1] 4.31
> mean(datos4[,2])
[1] 5.012128
```

si la anterior matriz de Rotaciones la denotamos como

```
Rotaciones:
     PC1  PC2
[1,]  a    c
[2,]  b    d
```

las ecuaciones de las componentes principales son, respectivamente,

$$x_2 = \overline{x}_2 - \frac{c}{d}(x_1 - \overline{x}_1)$$

$$x_2 = \overline{x}_2 - \frac{a}{b}(x_1 - \overline{x}_1)$$

Es decir,

$$x_2 = 5'01 - \frac{-0'99}{-0'14}(x_1 - 4'31)$$

$$x_2 = 5'01 - \frac{0'14}{-0'99}(x_1 - 4'31)$$

o bien,

$$x_2 = 35'48 - 7'07\, x_1$$

$$x_2 = 4'41 + 0'14\, x_1$$

rectas que añadimos a la nube de puntos mediante los siguientes comandos

```
> plot(datos4[,1],datos4[,2],xlim=c(3.4,6.5),ylim=c(3.4,6.5), pch=16)
> abline(35.48,-7.07,col=2)
> abline(4.41,0.14,col=2)
```

Hemos fijado con la primera sentencia una misma amplitud de los recorridos de ambas variables con objeto de que las componentes principales se muestren perpendiculares en el gráfico de la figura 2.

Como se observa en dicho gráfico, la primera componente principal se ve influenciada por los cuatro outliers y, por tanto, también la segunda al tener que ser perpendicular a la primera.

Si queremos realizar un Análisis de Componentes Principales robusto, supuesto que tenemos abierto el módulo `mva` y el módulo `lqs`, ejecutaremos `(5)` y `(6)` si queremos realizarlo utilizando como estimador robusto de la matriz de covarianzas, el proporcionado por el estimador elipsoide de mínimo volumen. En `(7)` se obtiene el centro del elipsoide y en `(8)` la matriz de covarianzas robusta a utilizar en el Análisis de Componentes Principales robusto.

```
> M<-cov.mve(datos4)                                          (5)
> M
$center
[1] 4.41275 4.93350                                           (7)
```

```
$cov
           [,1]       [,2]
[1,] 0.01150763 0.03851321                                    (8)
[2,] 0.03851321 0.24101308

$msg
[1] "16 singular samples of size 3 out of 1500"

$crit
[1] -3.91324

$best
 [1]  2  4  6 10 13 15 16 21 22 24 25 26 27 28 31 33 37 38 39 41 42 43
44 45 46

$n.obs
[1] 47

> robuscp(datos4,covmat=M)$loadings                           (6)
       Comp.1     Comp.2
[1,] 0.1611968  0.9869223
[2,] 0.9869223 -0.1611968
attr(,"class")
[1] "loadings"
```

A partir de ahí, las componentes principales robustas, utilizando el estimador del elipsoide de mínimo volumen tendrán por ecuaciones (empleando también el estimador robusto de localización suministrado por dicho estimador)

$$x_2 = 4'93 - \frac{0'987}{-0'161}(x_1 - 4'41)$$

$$x_2 = 4'93 - \frac{0'161}{0'987}(x_1 - 4'41)$$

es decir,

$$x_2 = -22'1 + 6'13\, x_1$$

$$x_2 = 5'65 - 0'163\, x_1$$

rectas que se añaden al gráfico 2 con las sentencias

```
> abline(-22.1,6.13,col=3,lty=2)
> abline(5.65,-0.163,col=3,lty=2)
```

donde se observa cómo estas componentes principales se ven mucho menos influenciadas por los outliers que las clásicas.

Ejecutando (9) y (10) obtendremos un Análisis de Componentes Principales robusto, utilizando el estimador suministrado por el método mcd,

```
> M<-cov.mcd(datos4)                                                    (9)
> M
$center
[1] 4.409024 4.949024

$cov
           [,1]       [,2]
[1,] 0.01178902 0.03517902
[2,] 0.03517902 0.24486902

$msg
[1] "12 singular samples of size 3 out of 1500"

$crit
[1] -8.031215

$best
 [1]  1  2  4  6  8 10 12 13 16 24 25 26 28 32 33 37 38 39 40 41 42 43
44 45 46

$n.obs
[1] 47

> robuscp(datos4,covmat=M)$loadings                                     (10)
        Comp.1     Comp.2
[1,] 0.1460578  0.9892761
[2,] 0.9892761 -0.1460578
```

```
attr(,"class")
[1] "loadings"
```

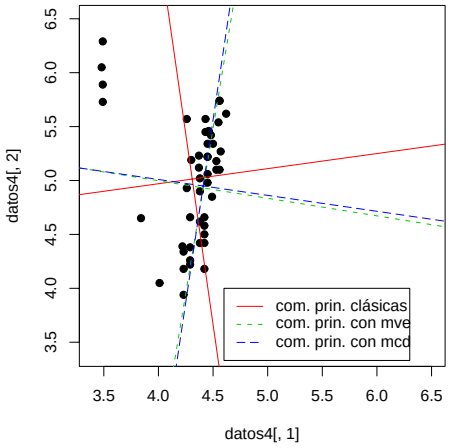

Figure 2: : Componentes principales clásicas y robustas del ejemplo

de donde se obtienen las componentes principales de ecuaciones,

$$x_2 = 4'95 - \frac{0'989}{-0'146}(x_1 - 4'41)$$

$$x_2 = 4'95 - \frac{0'146}{0'989}(x_1 - 4'41)$$

es decir,

$$x_2 = -24'92 + 6'774\, x_1$$

$$x_2 = 5'6 - 0'1476\, x_1$$

rectas que se añaden al gráfico 2 con los comandos

```
> abline(-24.92,6.774,col=4,lty=5)
> abline(5.6,-0.1476,col=4,lty=5)
> legend(4.7,4,c("clásico","mve","mcd"),lty=c(1,2,5),col=c(2,3,4))
```

componentes que no se diferencian mucho de las suministradas por el otro método robusto.

Aunque en este segundo caso, para distinguirlas, hemos cambiado el tipo de línea que las representa (`lty=5`), hemos añadido, con la función `legend`, un rótulo para diferenciar las componentes principales más claramente.