UNIVERSITAT DE BARCELONA

Universitat Oberta de Catalunya

Análisis de correspondencias
Soluciones

Francesc Carmona y Josep Gregori*

2 de marzo de 2021

Ejercicio 1

Tabla de frecuencias absolutas.

```
> tabla <- matrix(c(6,1,4,2,1,3,25,2,11,11,0,20),4,3)
> colnames(tabla) <- c("E1","E2","E3")
> rownames(tabla) <- c("A1","A2","A3","A4")
> tabla.N <- as.table(tabla)
> tabla.N

   E1 E2 E3
A1  6  1 11
A2  1  3 11
A3  4 25  0
A4  2  2 20
```

Tabla de frecuencias relativas.

```
> n <- sum(tabla.N)
> tabla.F <- tabla.N/n
> round(tabla.F,4)

       E1     E2     E3
A1 0.0698 0.0116 0.1279
A2 0.0116 0.0349 0.1279
A3 0.0465 0.2907 0.0000
A4 0.0233 0.0233 0.2326
```

Frecuencias relativas marginales.

```
> margin.f <- apply(tabla.F,1,sum)
> margin.c <- apply(tabla.F,2,sum)
```

o también con `margin.table()`

```
> margin.f <- margin.table(tabla.F,1)
> margin.c <- margin.table(tabla.F,2)
```

La tabla de frecuencias relativas con marginales:

*Alumno del curso 2009-10

```
> round(addmargins(tabla.F),4)

        E1     E2     E3    Sum
A1  0.0698 0.0116 0.1279 0.2093
A2  0.0116 0.0349 0.1279 0.1744
A3  0.0465 0.2907 0.0000 0.3372
A4  0.0233 0.0233 0.2326 0.2791
Sum 0.1512 0.3605 0.4884 1.0000
```

Frecuencias relativas condicionadas por filas.

```
> tabla.P <- diag(1/margin.f) %*% tabla.F
> tabla.P <- sweep(tabla.F, 1, margin.f, "/") # más eficiente
```

o directamente de la `tabla.N` de frecuencias absolutas:

```
> perfiles <- prop.table(tabla.N, 1)
```

Representación en dimensión 3 con el paquete plotly.

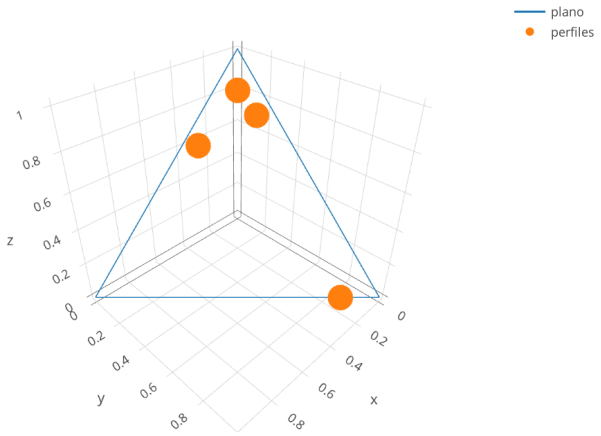


Figura 1: Perfiles de las cuatro regiones en el espacio tridimensional.

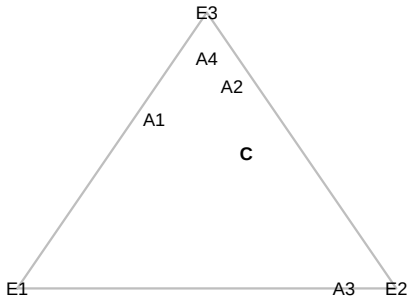
Una vez instalado el paquete plotly ya podemos dibujar los perfiles:

```
> library(plotly)
> graph3d <- plot_ly( x = c( 1, 0, 0, 1 ) ,
+                     y = c( 0, 1, 0, 0 ) ,
+                     z = c( 0, 0, 1, 0 ) ,
+                     type = 'scatter3d' , mode = 'lines' ,
+                     name = 'plano') %>%
+           add_trace( x = perfiles[,1] ,
+                      y = perfiles[,2] ,
+                      z = perfiles[,3] ,
+                      type = 'scatter3d' , mode = 'markers',
+                      name = 'perfiles')
> graph3d
```

El resultado es un gráfico tridimensional como el de la figura 1 que se puede mover en un navegador.

Representación bidimensional.

```
> tabla.x <- 1 - perfiles[,1] - perfiles[,3]/2
> tabla.y <- perfiles[,3] * sqrt(3)/2
> plot.new()
> lines(c(0,1,0.5,0),c(0,0,sqrt(3)/2,0), col="gray", lwd=2)
> text(c(0,1,0.5),c(0,0,sqrt(3)/2),labels=colnames(tabla))
> text(tabla.x,tabla.y,labels=rownames(tabla))
> # Punto medio o centroide de las filas (es la marginal de las columnas)
> text(1-margin.c[1]-margin.c[3]/2,margin.c[3]*sqrt(3)/2,labels=c("C"),font=2)
```



También es posible añadir el centroide al gráfico 3D que hemos hecho con el paquete plotly.

```
> graph3d %>%
+    add_trace( x = margin.c[1] ,
+               y = margin.c[2] ,
+               z = margin.c[3] ,
+            type = 'scatter3d' , mode = 'markers',
+            name = 'centroide')
```

El centroide también queda en el mismo plano.

Ejercicio 2

Distancia ji-cuadrado entre filas.

```
> nf <- nrow(tabla.N)
> D2.chisq <- matrix(0,nf,nf)
> for(i in 1:(nf-1))
+ for(j in i:nf)
+ D2.chisq[i,j] <-
+   t(tabla.P[i,]-tabla.P[j,]) %*% diag(1/margin.c) %*% (tabla.P[i,]-tabla.P[j,])
> D2.chisq <- D2.chisq + t(D2.chisq)
> rownames(D2.chisq) <- colnames(D2.chisq) <- rownames(tabla.N)
> #
> D2.chisq
```

```
          A1         A2       A3         A4
A1 0.0000000 0.55889656 2.821799 0.51671911
A2 0.5588966 0.00000000 2.350788 0.06007365
A3 2.8217989 2.35078798 0.000000 3.12402946
A4 0.5167191 0.06007365 3.124029 0.00000000
```

Distancia de las filas a su centroide:

```
> dc <-  vector(mode="numeric",nf)
> for(i in 1:nf)
+ dc[i] <- t(tabla.P[i,]-margin.c) %*% diag(1/margin.c) %*% (tabla.P[i,]-margin.c)
> names(dc) <- rownames(tabla.N); dc

       A1        A2        A3        A4
0.5083023 0.2415335 1.1875352 0.4871631
```

Inercia total:

```
> as.numeric(dc %*% margin.f)

[1] 0.6849172
```

coincide con la inercia total con la función `chisq.test()`:

```
> as.numeric(chisq.test(tabla.N)$stat)/n

[1] 0.6849172
```

La distancia ji-cuadrado entre perfiles equivale a la distancia euclídea entre los vectores transformados $\mathbf{y}_i$:

```
> Y <- tabla.P %*% diag(1/sqrt(margin.c))
> colnames(Y) <- colnames(tabla.N); Y

          E1         E2        E3
A1 0.8573463 0.09253284 0.8744697
A2 0.1714693 0.33311821 1.0493636
A3 0.3547640 1.43585435 0.0000000
A4 0.2143366 0.13879925 1.1924587

> # Transformación del centroide
> cc <- margin.c %*% diag(1/sqrt(margin.c)); cc

          [,1]      [,2]      [,3]
[1,] 0.3887966 0.6003875 0.6988362

> # Distancias euclídeas al cuadrado entre vectores transformados y su centroide
> as.matrix(dist(rbind(cc,Y)))[1,2:5]^2

       A1        A2        A3        A4
0.5083023 0.2415335 1.1875352 0.4871631
```

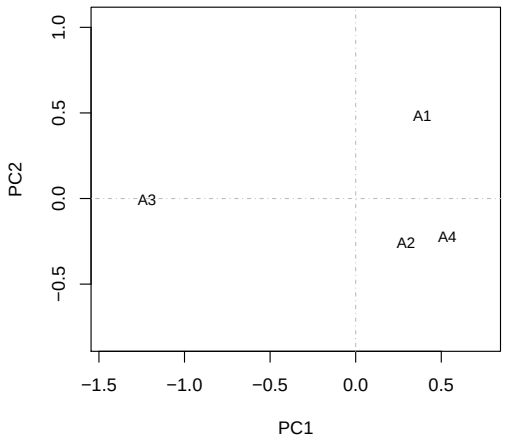
Ejercicio 3

Escalado multidimensional sobre la matriz de distancias ji-cuadrado entre filas.

```
> mds <- cmdscale(sqrt(D2.chisq),eig=TRUE); mds

$points
         [,1]         [,2]
A1  0.3888672  0.483339233
A2  0.2937117 -0.258174080
A3 -1.2184463 -0.004863786
A4  0.5358673 -0.220301367

$eig
[1]  2.009249e+00  3.488270e-01 -4.598783e-17 -1.110223e-16

$x
NULL

$ac
[1] 0

$GOF
[1] 1 1
```

Las siguientes instrucciones representan el mapa de distancias ji-cuadrado.

```
> library(MASS)
> eqscplot(mds$points,ty="n",xlab="PC1",ylab="PC2",xlim=c(-1.5,0.8))
> abline(v=0,h=0, col="gray",lty=4)
> text(mds$points[,1],mds$points[,2],labels=rownames(tabla.N),cex=0.8)
```



Ahora vamos a representar las columnas. Para ello necesitamos las distancias ji-cuadrado entre columnas.

```
> # Frecuencias relativas condicionadas por columnas
> tabla.Pc <- tabla.F %*% diag(1/margin.c); tabla.Pc
```

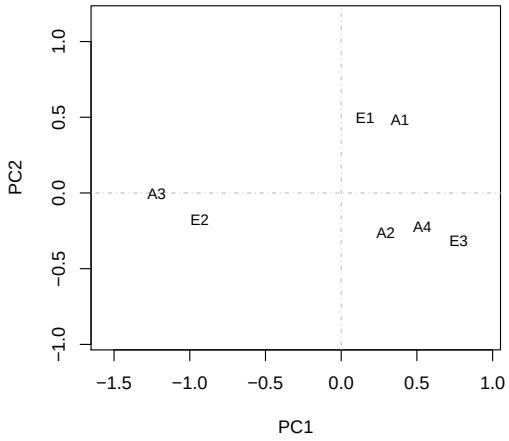
```
         [,1]       [,2]      [,3]
A1 0.46153846 0.03225806 0.2619048
A2 0.07692308 0.09677419 0.2619048
A3 0.30769231 0.80645161 0.0000000
A4 0.15384615 0.06451613 0.4761905

> colnames(tabla.Pc) <- colnames(tabla.N)
> # Distancia ji-cuadrado entre columnas
> nc <- ncol(tabla.N)
> D2c.chisq <- matrix(0,nc,nc)
> for(i in 1:(nc-1))
+ for(j in i:nc)
+ D2c.chisq[i,j] <-
+ t(tabla.Pc[,i]-tabla.Pc[,j]) %*% diag(1/margin.f) %*% (tabla.Pc[,i]-tabla.Pc[,j])
> D2c.chisq <- D2c.chisq + t(D2c.chisq);
> rownames(D2c.chisq ) <- colnames(D2c.chisq) <- colnames(tabla.N); D2c.chisq

         E1       E2       E3
E1 0.000000 1.649015 1.039685
E2 1.649015 0.000000 2.944260
E3 1.039685 2.944260 0.000000
```

Ahora calculamos el **escalado multidimensional sobre la matriz de distancias entre columnas**.

```
> #
> mds.c <- cmdscale(sqrt(D2c.chisq),eig=TRUE); mds.c

$points
         [,1]       [,2]
E1  0.1586441  0.4951574
E2 -0.9344939 -0.1786857
E3  0.7758497 -0.3164717

$eig
[1] 1.500390e+00 3.772638e-01 1.554312e-15

$x
NULL

$ac
[1] 0

$GOF
[1] 1 1
```

Así ya podemos representar filas y columnas.

```
> eqscplot(mds.c$points,ty="n",xlab="PC1",ylab="PC2",xlim=c(-1.6,1),ylim=c(-0.6,0.8))
> abline(v=0,h=0, col="gray",lty=4)
> text(mds$points[,1],mds$points[,2],labels=rownames(tabla.N),cex=0.8)
> text(mds.c$points[,1],mds.c$points[,2],labels=colnames(tabla.N),cex=0.8)
```



Ejercicio 4

Análisis de componentes principales (con la descomposión en valores singulares).

```
> Df <- diag(margin.f)
> Dc <- diag(margin.c)
> Dfmh <- diag(1/sqrt(margin.f))
> Dcmh <- diag(1/sqrt(margin.c))
> Z <- Dfmh %*% (tabla.F - margin.f %o% margin.c) %*% Dcmh
> Z.svd <- svd(Z)
```

Las coordenadas principales (pc) y estándares (sc) son:

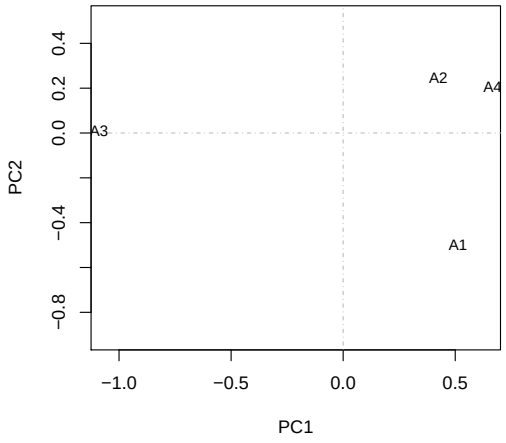
```
> filas.sc  <- Dfmh %*% Z.svd$u
> cols.sc <- Dcmh %*% Z.svd$v
> filas.pc <- filas.sc %*% diag(Z.svd$d)
> cols.pc <- cols.sc %*% diag(Z.svd$d)
```

y las inercias

```
> inercias <- Z.svd$d^2
> # inercia total
> sum(inercias)

[1] 0.6849172
```

El mapa de distancias entre las filas es

```
> eqscplot(-filas.pc[,1:2],type="n",xlab="PC1",ylab="PC2",ylim=c(-0.8,0.4))
> abline(v=0,h=0, col="gray",lty=4)
> text(-filas.pc[,1],-filas.pc[,2],labels=rownames(tabla.N),cex=0.8)
```



Ejercicio 5

La tabla 1 muestra los colores de pelo y de ojos de un gran número de personas.

```
> # Taula de freqüències absolutes
> eyes.hair <- matrix(c(688,326,343,98,116,38,84,48,584,241,909,403,
+                       188,110,412,681,4,3,26,81),4,5)
> colnames(eyes.hair) <- c("fair","red","medium","dark","black")
> rownames(eyes.hair) <- c("light","blue","medium","dark")
> eyes.hair <- as.table(eyes.hair); eyes.hair

       fair red medium dark black
light   688 116    584  188     4
blue    326  38    241  110     3
medium  343  84    909  412    26
dark     98  48    403  681    81
```

Hallar la solución bidimensional del análisis de correspondencias.

(a) *Como un escalado multidimensional de filas y de columnas con la distancia ji-cuadrado.*

La matriz de frecuencias relativas y vectores marginales son:

```
> tabla.F <- eyes.hair/sum(eyes.hair)
> margin.f  <- margin.table(tabla.F,1)
> margin.c <- margin.table(tabla.F,2)
```

La matriz de frecuencias relativas condicionada por filas es:

```
> tabla.P <- diag(1/margin.f) %*% tabla.F
> rownames(tabla.P) <- rownames(tabla.F); tabla.P

            fair        red    medium      dark       black
light  0.4354430 0.07341772 0.3696203 0.1189873 0.002531646
blue   0.4540390 0.05292479 0.3356546 0.1532033 0.004178273
medium 0.1933484 0.04735062 0.5124014 0.2322435 0.014656144
dark   0.0747521 0.03661327 0.3073989 0.5194508 0.061784897
```

La distancia ji-cuadrado entre filas.

```
> nf <- nrow(tabla.F)
> D2.chisq <- matrix(0,nf,nf)
> for(i in 1:(nf-1))
+ for(j in i:nf)
+ D2.chisq[i,j] <-
+   t(tabla.P[i,]-tabla.P[j,]) %*% diag(1/margin.c) %*% (tabla.P[i,]-tabla.P[j,])
> D2.chisq <- D2.chisq + t(D2.chisq)
> rownames(D2.chisq) <- colnames(D2.chisq) <- rownames(tabla.F); D2.chisq

            light       blue    medium      dark
light  0.00000000 0.01674839 0.3375581 1.3029657
blue   0.01674839 0.00000000 0.3600631 1.2150390
medium 0.33755808 0.36006307 0.0000000 0.5841656
dark   1.30296565 1.21503900 0.5841656 0.0000000
```

Escalado multidimensional sobre la matriz de distancias ji-cuadrado entre filas.

```
> mds <- cmdscale(sqrt(D2.chisq),eig=TRUE); mds

$points
              [,1]        [,2]
light  -0.41443523  0.04093269
blue   -0.37720215  0.12328994
medium  0.06772094 -0.27789378
dark    0.72391643  0.11367114

$eig
[1] 8.426791e-01 1.070220e-01 4.433835e-03 4.998716e-17

$x
NULL

$ac
[1] 0

$GOF
[1] 0.995353 0.995353
```

Ahora para las columnas:

```
> tabla.Pc <- tabla.F %*% diag(1/margin.c)
> colnames(tabla.Pc) <- colnames(tabla.F); tabla.Pc

             fair       red    medium      dark      black
light  0.47285223 0.4055944 0.2732803 0.1351546 0.03508772
blue   0.22405498 0.1328671 0.1127749 0.0790798 0.02631579
medium 0.23573883 0.2937063 0.4253627 0.2961898 0.22807018
dark   0.06735395 0.1678322 0.1885821 0.4895758 0.71052632

> # Distancia ji-cuadrado entre columnas
> nc <- ncol(tabla.F)
```

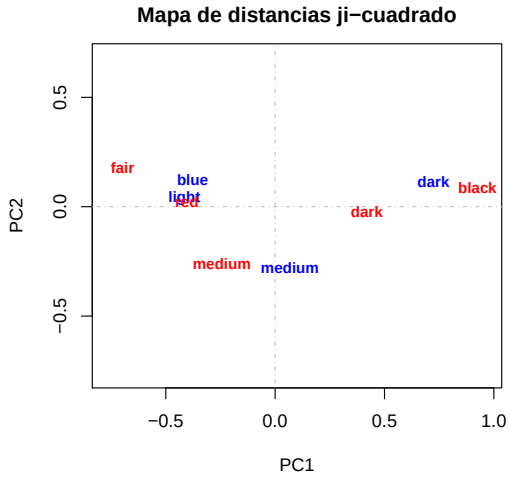
```
> D2c.chisq <- matrix(0,nc,nc)
> for(i in 1:(nc-1))
+ for(j in i:nc)
+ D2c.chisq[i,j] <-
+   t(tabla.Pc[,i]-tabla.Pc[,j]) %*% diag(1/margin.f) %*% (tabla.Pc[,i]-tabla.Pc[,j])
> D2c.chisq <- D2c.chisq + t(D2c.chisq);
> rownames(D2c.chisq ) <- colnames(D2c.chisq) <- colnames(tabla.F); D2c.chisq

            fair       red    medium      dark     black
fair   0.0000000 0.1294029 0.3979869 1.2891793 2.6447706
red    0.1294029 0.0000000 0.1170365 0.6959378 1.7751753
medium 0.3979869 0.1170365 0.0000000 0.4961377 1.4860380
dark   1.2891793 0.6959378 0.4961377 0.0000000 0.2695207
black  2.6447706 1.7751753 1.4860380 0.2695207 0.0000000
```

Escalado multidimensional sobre la matriz de distancias ji-cuadrado entre columnas.

```
> mds.c <- cmdscale(sqrt(abs(D2c.chisq)),eig=TRUE); mds.c

$points
             [,1]        [,2]
fair   -0.6975450  0.17672578
red    -0.4036977  0.02229924
medium -0.2435867 -0.26024708
dark    0.4198572 -0.02426160
black   0.9249723  0.08548366

$eig
[1]  1.740729e+00  1.073539e-01  1.215408e-02  1.595297e-17 -1.387779e-17

$x
NULL

$ac
[1] 0

$GOF
[1] 0.9934664 0.9934664
```

Representación gráfica de filas y columnas:

```
> eqscplot(mds.c$points,ty="n",xlab="PC1",ylab="PC2",xlim=c(-0.8,1))
> abline(v=0,h=0, col="gray",lty=4)
> text(mds$points[,1],mds$points[,2],labels=rownames(tabla.F),cex=0.8,font=2,col="blue")
> text(mds.c$points[,1],mds.c$points[,2],labels=colnames(tabla.F),cex=0.8,
+                                        font=2,col= "red")
> title(main="Mapa de distancias ji-cuadrado",line=1)
```



(b) *Como un análisis de componentes principales sobre la matriz* **Z** *estandarizada.*

La matriz **Z** estandarizada es:

```
> Z <- diag(1/sqrt(margin.f)) %*% (tabla.F - margin.f %*% t(margin.c)) %*% diag(1/sqrt(margin.c)
> rownames(Z) <- rownames(tabla.F)
> colnames(Z) <- colnames(tabla.F); round(Z,4)

          fair     red  medium    dark   black
light   0.1721  0.0477 -0.0235 -0.1486 -0.0694
blue    0.1291 -0.0003 -0.0356 -0.0756 -0.0427
medium -0.0850 -0.0144  0.1052 -0.0295 -0.0257
dark   -0.1856 -0.0354 -0.0702  0.2534  0.1377
```

Análisis de componentes principales con la descomposición SVD:

```
> Z.svd <- svd(Z); Z.svd

$d
[1] 4.449117e-01 1.727295e-01 2.917206e-02 2.635591e-17

$u
            [,1]       [,2]        [,3]      [,4]
[1,] -0.53575701 -0.2734448 -0.58709090 0.5417717
[2,] -0.32799355 -0.3483142  0.79857033 0.3652162
[3,]  0.04661383  0.8100541  0.10992499 0.5740697
[4,]  0.77666711 -0.3843404 -0.07433838 0.4935023

$v
            [,1]        [,2]        [,3]       [,4]
[1,] -0.63531413 -0.51817340  0.22276701 -0.4621795
[2,] -0.12042064 -0.06365097 -0.93268485 -0.2968492
[3,] -0.05693066  0.75822910  0.07544383 -0.5501921
[4,]  0.67395266 -0.31481695  0.16417879 -0.6117099
[5,]  0.35273439 -0.23113589 -0.21869472  0.1461858
```

Inercia total a partir de los valores propios de la SVD:

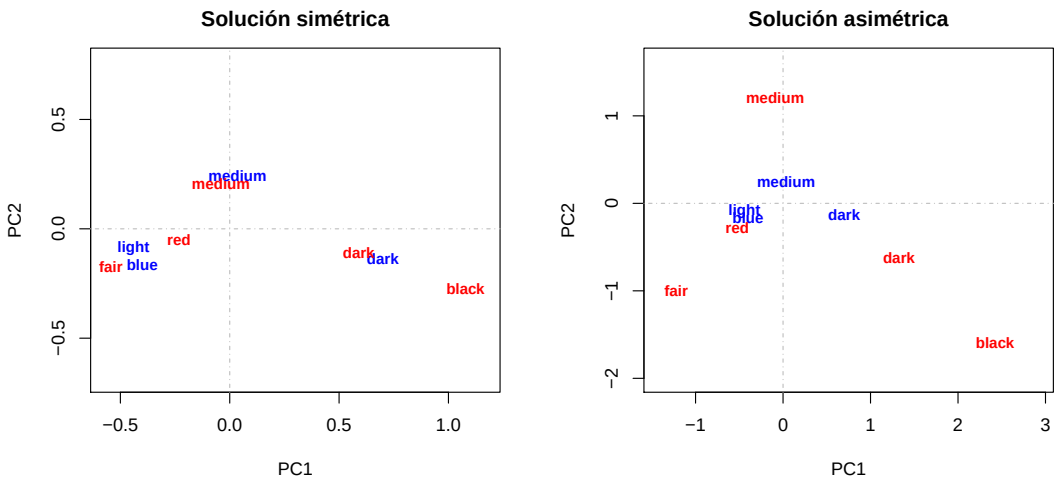
```
> sum(Z.svd$d^2)

[1] 0.2286329
```

Coordenadas estandar y principales

```
> f.sc <- diag(1/sqrt(margin.f)) %*% Z.svd$u
> c.sc <- diag(1/sqrt(margin.c)) %*% Z.svd$v
> f.pc <- f.sc %*% diag(Z.svd$d)
> c.pc <- c.sc %*% diag(Z.svd$d)
```

Mapas de distancias con filas y columnas de **Z** (solución simétrica y asimétrica):

```
> # solución simétrica
> eqscplot(f.pc[,1:2],type="n",xlab="PC1",ylab="PC2",xlim=c(-0.6,1.2))
> abline(v=0,h=0, col="gray",lty=4)
> text(f.pc[,1],f.pc[,2],labels=rownames(tabla.F),cex=0.8,font=2,col="blue")
> text(c.pc[,1],c.pc[,2],labels=colnames(tabla.F),cex=0.8,font=2,col= "red")
> title(main="Solución simétrica",line=1)
> #
> # solución asimétrica
> eqscplot(c.sc[,1:2],type="n",xlab="PC1",ylab="PC2",xlim=c(-1.5,3))
> abline(v=0,h=0, col="gray",lty=4)
> text(f.pc[,1],f.pc[,2],labels=rownames(tabla.F),cex=0.8,font=2,col="blue")
> text(c.sc[,1],c.sc[,2],labels=colnames(tabla.F),cex=0.8,font=2,col= "red")
> title(main="Solución asimétrica",line=1)
```



(c) *Con la función* `ca()` *del paquete* `ca` *de* ***R****.*

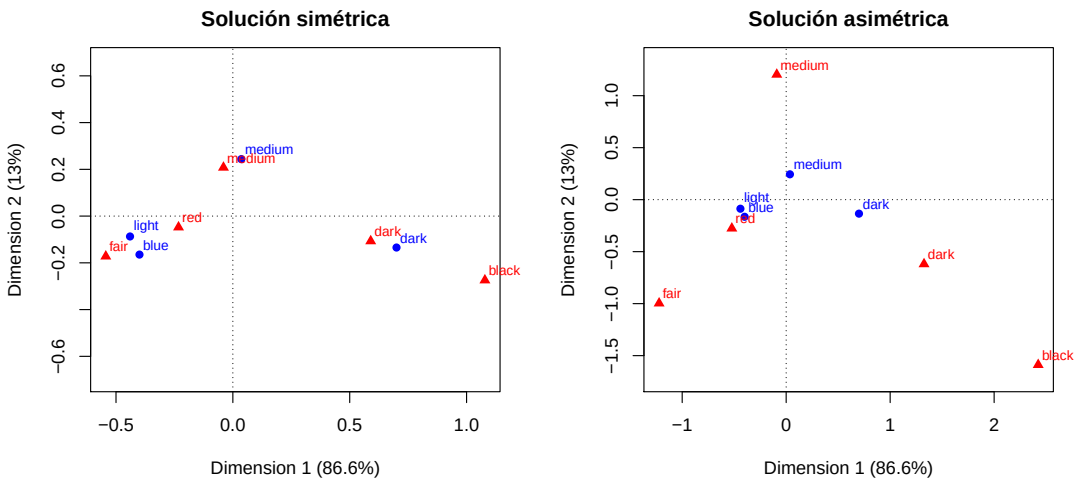
Un `plot()` *del resultado proporciona la representación en dos dimensiones.*

```
> library(ca)
> ca(tabla.F)
```

```
 Principal inertias (eigenvalues):
           1        2        3
Value      0.197946 0.029835 0.000851
Percentage 86.58%   13.05%   0.37%


 Rows:
            light      blue   medium      dark
Mass     0.293517  0.133383 0.329556  0.243544
ChiDist  0.449639  0.436875 0.246460  0.713015
Inertia  0.059342  0.025457 0.020018  0.123816
Dim. 1  -0.988898 -0.898081 0.081199  1.573786
Dim. 2  -0.504723 -0.953721 1.411073 -0.778802


 Columns:
             fair       red    medium      dark     black
Mass     0.270295  0.053130  0.396991  0.258406  0.021178
ChiDist  0.570422  0.265019  0.211743  0.599559  1.113616
Inertia  0.087949  0.003732  0.017799  0.092890  0.026263
Dim. 1  -1.221994 -0.522432 -0.090356  1.325800  2.423860
Dim. 2  -0.996680 -0.276143  1.203401 -0.619308 -1.588280
```

Representaciones simétrica y asimétrica:

```
> # solución simétrica
> plot(ca(tabla.F))
> title(main="Solución simétrica",line=1)
> # solución asimétrica
> plot(ca(tabla.F),map= "rowprincipal")
> title(main="Solución asimétrica",line=1)
```



Ejercicio 6

La tabla `smoke` *del paquete* `ca` *contiene la clasificación de los empleados de una empresa según su nivel profesional (cinco grupos) y sus hábitos fumadores (cuatro grupos).*

```
> data(smoke, package = "ca")
> smoke

   none light medium heavy
SM    4     2      3     2
JM    4     3      7     4
SE   25    10     12     4
JE   18    24     33    13
SC   10     6      7     2
```

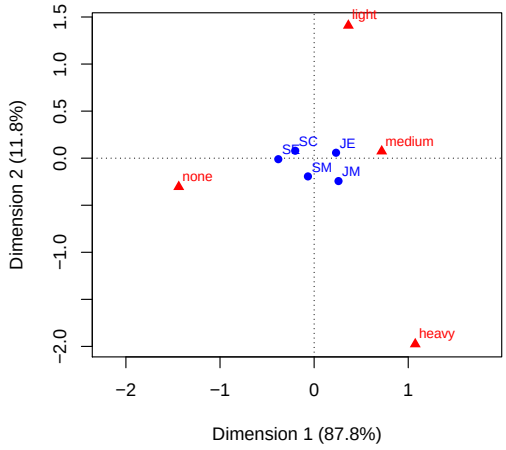
(a) *Dibujar un mapa óptimo del AC bidimensional y asimétrico, con las filas en coordenadas principales (proyecciones de los perfiles) y las columnas en coordenadas estándares (proyecciones de los vértices).*

El mapa asimétrico se puede conseguir con la opción `map="rowprincipal"` *de la función* `plot()` *para un* `ca`.

```
> (smoke.ca <- ca(smoke))


 Principal inertias (eigenvalues):
           1        2        3
Value      0.074759 0.010017 0.000414
Percentage 87.76%   11.76%   0.49%


 Rows:
               SM        JM        SE       JE        SC
Mass     0.056995  0.093264  0.264249 0.455959  0.129534
ChiDist  0.216559  0.356921  0.380779 0.240025  0.216169
Inertia  0.002673  0.011881  0.038314 0.026269  0.006053
Dim. 1  -0.240539  0.947105 -1.391973 0.851989 -0.735456
Dim. 2  -1.935708 -2.430958 -0.106508 0.576944  0.788435


 Columns:
             none    light   medium     heavy
Mass     0.316062 0.233161 0.321244  0.129534
ChiDist  0.394490 0.173996 0.198127  0.355109
Inertia  0.049186 0.007059 0.012610  0.016335
Dim. 1  -1.438471 0.363746 0.718017  1.074445
Dim. 2  -0.304659 1.409433 0.073528 -1.975960
```

Mapa asimétrico con las filas en CP:

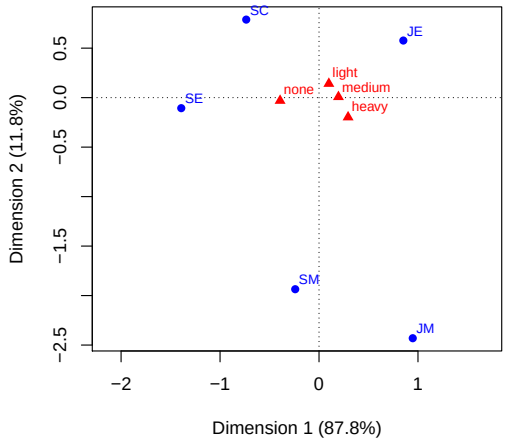
```
> plot(smoke.ca, map="rowprincipal")
```



(b) *Dibujar un mapa asimétrico, con las columnas en coordenadas principales y las filas en coordenadas estándares.*

Mapa asimétrico con las columnas en CP:

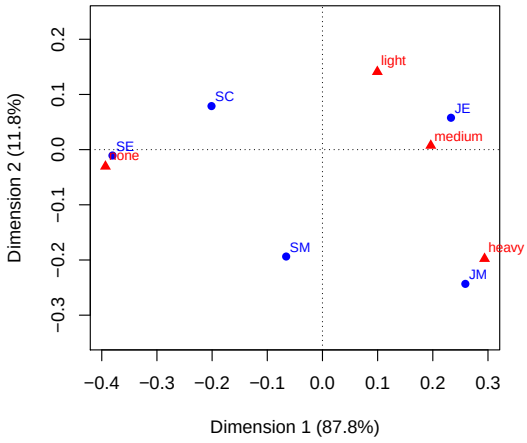
```
> plot(smoke.ca, map="colprincipal")
```



(c) *Dibujar un mapa simétrico de los datos sobre los hábitos de los fumadores, es decir, representar tanto las filas como las columnas en coordenadas principales.*

Mapa simétrico con las columnas en CP

```
> plot(smoke.ca)
```



Ejercicio 7

El AC se utiliza ampliamente para analizar datos en ecología. Los datos del archivo `benthos.xls` *que se pueden hallar en la web* `www.carme-n.org` *corresponden a los recuentos de 92 especies marinas identificadas en 13 muestras del fondo marino del mar del Norte. La mayor parte de las muestras se obtuvieron cerca de una plataforma petrolífera que producía una cierta contaminación del fondo marino. Existen dos muestras, utilizadas como referencia, supuestamente no contaminadas, que se obtuvieron lejos de la zona de influencia de la plataforma petrolífera.*

```
> library(readxl)
> benthos <- as.data.frame(read_excel("benthos.xls", sheet = "92EKOBIO"))
> rownames(benthos) <- benthos$...1
> benthos <- benthos[,-1]
> head(benthos)

           S4 S8  S9 S12 S13 S14 S15 S18 S19 S23 S24 R40 R42
Myri_ocul 193 79 150  72 141 302 114 136 267 271 992   5  12
Chae_seto  34  4 247  19  52 250 331  12 125  37  12   8   3
Amph_falc  49 58  66  47  78  92 113  38  96  76  37   0   5
Myse_bide  30 11  36  65  35  37  21   3  20 156  12  58  43
Goni_macu  35 39  41  37  32  45  41  41  31  29  64  32  23
Amph_fili  19 39  11  38  18  20  11  22  30  40   3  55  65
```

(a) *Calcular la inercia total.*

La inercia total es:

```
> as.numeric(chisq.test(benthos)$stat)/sum(benthos)

[1] 0.7826499
```

De la descomposición SVD de `ca()`:

```
> sum(ca(benthos)$rowinertia)

[1] 0.7826499
```

¡Atención! En este caso no podemos aplicar el test χ^2, aunque sí podemos calcular la inercia, ya que los datos no son una verdadera tabla de contingencia. Cada observación no es independiente de las otras puesto que los organismos marinos se presentan agrupados en un determinado punto muestral.

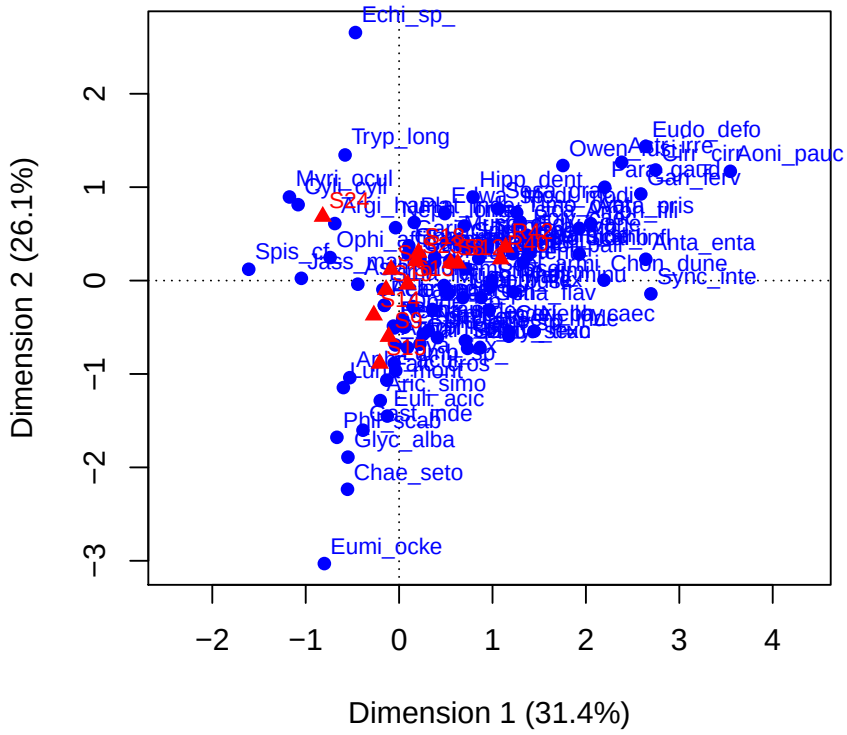
(b) *Representar los datos en un mapa asimétrico con las estaciones de muestreo en coordenadas principales y las especies en coordenadas estándares, es decir, el mapa asimétrico de los perfiles de las muestras (columnas) y de los vértices de las especies (filas).*

Análisis de correspondencias.

```
> benthos.ca <- ca(benthos); benthos.ca


 Principal inertias (eigenvalues):
           1        2        3        4        5        6        7
Value      0.245741 0.204369 0.126105 0.056241 0.031202 0.025107 0.022041
Percentage 31.4%    26.11%   16.11%   7.19%    3.99%    3.21%    2.82%
           8        9        10       11       12
Value      0.021378 0.015653 0.013435 0.011735 0.009642
Percentage 2.73%    2%       1.72%    1.5%     1.23%
...
```

Mapa asimétrico con las columnas en CP.

```
> plot(benthos.ca, map="colprincipal")
```



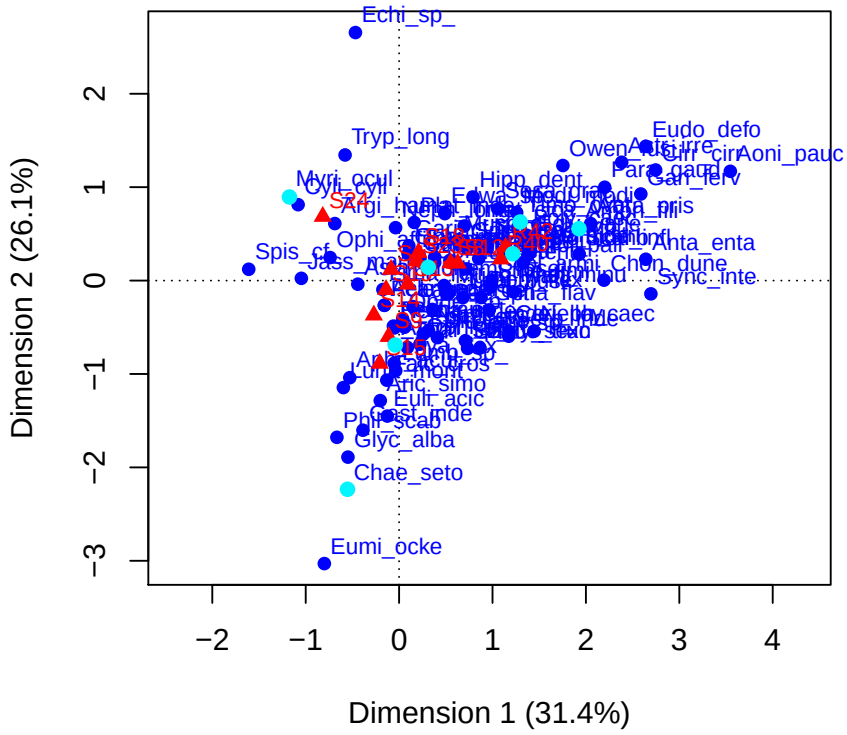
(c) *Identificar en el mapa las 7 especies más abundantes e interpretar los resultados en cuanto a las muestras y la contaminación.*

Las 7 especies más abundantes son:

```
> (seven <- names(sort(apply(benthos,1,sum),decreasing=TRUE))[1:7])

[1] "Myri_ocul" "Chae_seto" "Amph_falc" "Myse_bide" "Goni_macu" "Amph_fili"
[7] "Timo_ovat"
```

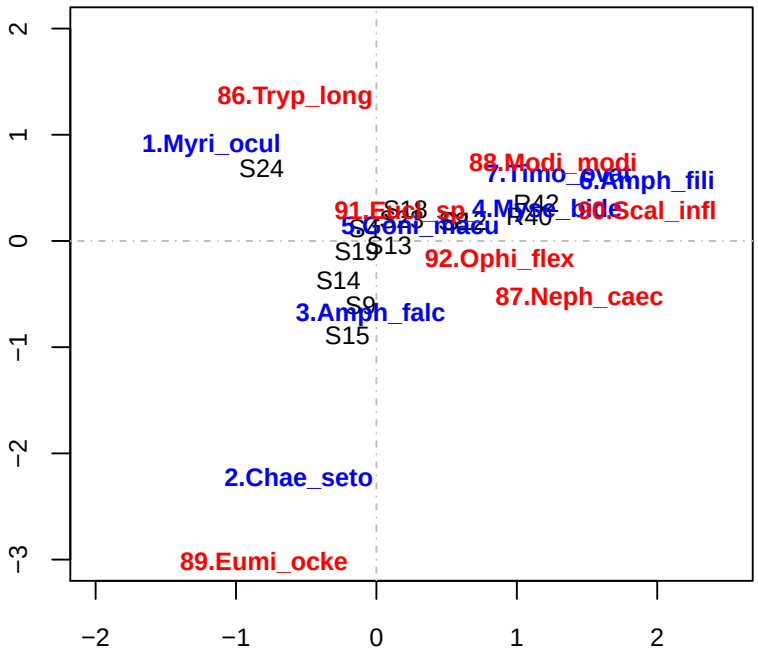
Ahora las destacamos en el mapa:

```
> aux <- which(seven %in% rownames(benthos))
> plot(benthos.ca, map="colprincipal")
> points(benthos.ca$rowcoord[aux,1:2],pch=19,col="turquoise1")
```



Para interpretar los resultados vamos a dibujar un mapa más simple, únicamente con las especies más abundantes (en azul) y las menos abundantes (en rojo).

```
> # abundants (en vermell)
> col.cp <- benthos.ca$colcoord %*% diag(benthos.ca$sv)
> row.ec <- benthos.ca$rowcoord
> par(cex.main=1,cex.lab=0.8,cex.axis=0.8)
> plot(col.cp[,1:2],type="n",ylim=c(-3,2),xlim=c(-2,2.5),xlab="",ylab="")
> abline(h=0,v=0,lty=4,col="gray")
> text(col.cp[,1],col.cp[,2],labels=colnames(benthos),cex=0.8)
> # 7 especies más abundantes
> nms <- paste(1:7,rownames(benthos)[aux],sep=".")
> text(row.ec[aux,1],row.ec[aux,2],labels=nms,cex=0.8,col="blue",font=2)
> # 7 especies menos abundantes
> aux2 <- order(apply(benthos,1,sum))[1:7]
> nml <- paste(seq(92,86,-1),rownames(benthos)[aux2],sep=".")
> text(row.ec[aux2,1],row.ec[aux2,2],labels=nml,cex=0.8,col="red",font=2)
```



En el gráfico se representan les 7 especies más abundantes, en azul, junto con las 7 menos abundantes, en rojo, y las 13 muestras, donde la R40 y la R42 son las que no estan bajo la influencia de una plataforma petrolífera. La numeración de les etiquetas da el orden decreciente de abundancia de las especies.

En esta representación observamos algunas cosas interesantes. La primera es que las estaciones se distribuyen a lo largo del primer eje según un orden decreciente en abundancia:

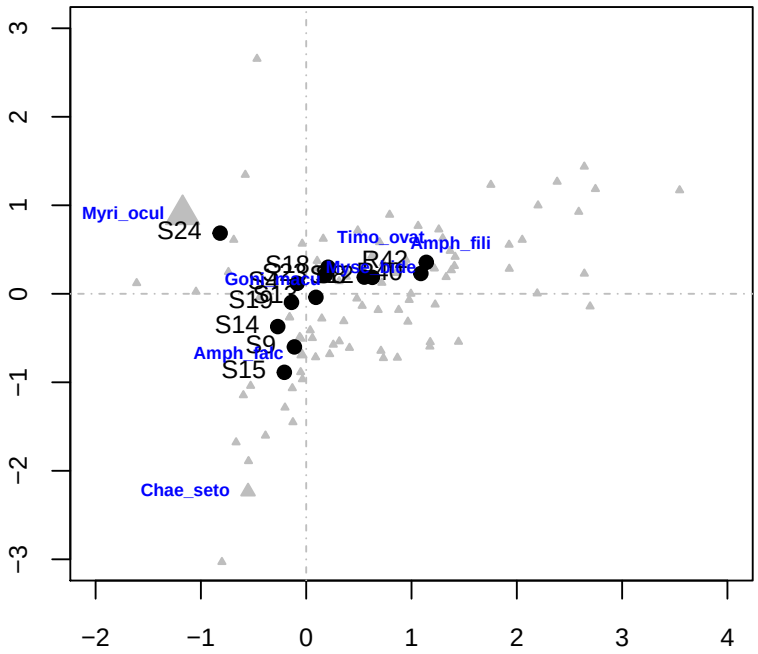
```
> sort(apply(benthos,2,sum),decreasing=TRUE)

 S24  S14  S23  S19  S15   S9  S12  S13   S4   S8  S18  R40  R42
1331 1043  978  888  871  827  658  644  594  577  516  355  313
```

También se observa la agrupación de las estaciones contaminadas, por un lado, y las no contaminadas por otro. La estación S24 está un poco alejada de todas y eso indica un comportamiento especial.

Las especies más abundantes siguen el mismo patrón y se distribuyen también a lo largo del primer eje. Las especies menos abundantes tienen una posición menos determinada.

```
> par(cex.main=1,cex.lab=0.8,cex.axis=0.8)
> plot(col.cp[,1:2],type="n",ylim=c(-3,3),xlim=c(-2,4),xlab="",ylab="")
> points(row.ec[,1],row.ec[,2],cex=0.4,col="grey",pch=24,bg="grey")
> points(row.ec[aux,1],row.ec[aux,2],cex=apply(benthos,1,sum)/1500,col="grey",pch=24,bg="grey")
> abline(h=0,v=0,lty=4,col="gray")
> points(col.cp[,1],col.cp[,2], pch=19)
> text(col.cp[,1],col.cp[,2],labels=colnames(benthos),pos=2,cex=0.8)
```

```
> # 7 especies más abundantes
> nms <- rownames(benthos)[aux]
> text(row.ec[aux,1],row.ec[aux,2],labels=nms,cex=0.6,col="blue",font=2,pos=2)
```



Sin contar la estación S24, caracterizada por la abundancia de la especie `Myri_ocul`, se ve que el resto de especies (como puntos) se agrupan formando una curva que arranca aproximadamente de abajo a la izquierda en fuerte pendiente positiva, pasa por el origen y ya en el primer cuadrante modera súbitamente la pendiente. En esa última zona se encuentran las muestras no contaminadas y a su alrededor las especies que las caracterizan.

En el gráfico anterior y para las 7 especies más abundantes, el tamaño del símbolo que las representa es proporcional a la abundancia.

Ejercicio 8

Recordemos los datos de los 24 meses observados por Florence Nightingale que pueden obtenerse en la página

`http://understandinguncertainty.org/node/214`

donde los 12 primeros son antes de aplicar sus nuevos métodos de cuidado en los hospitales militares. Consideremos las frecuencias de muertes por tres causas: Zymotic diseases, Wounds & injuries y All other causes, junto con la cuarta categoría de soldados en activo que se obtiene al restar los soldados muertos por alguna causa del total.
Con esa tabla de contingencia realizar un análisis de correspondencias completo y valorar e interpretar el resultado.

En primer lugar vamos a cargar los datos y prepararlos.

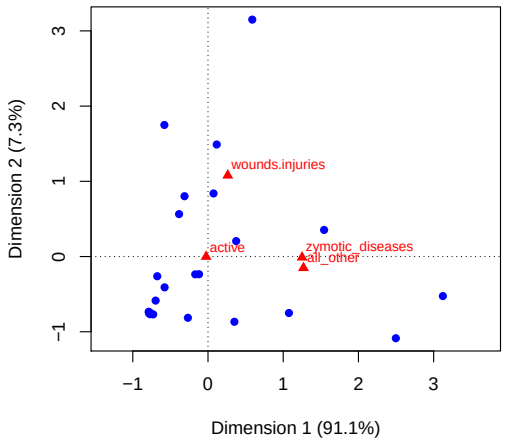
```
> ntg <- read.csv("nigthtingale.csv", sep=";")
> #ntg$treatment <- factor(c(rep(0,12),rep(1,12)))
> #levels(ntg$treatment) <- c("No", "Yes")
> ntg$active <- ntg$size - (ntg$zymotic_diseases + ntg$wounds.injuries + ntg$all_other)
```

El análisis de correspondencias es:

```
> ntg.ca <- ca(ntg[,5:8]); ntg.ca


 Principal inertias (eigenvalues):
           1        2        3
Value      0.031117 0.002489 0.000561
Percentage 91.07%   7.28%    1.64%
...
```

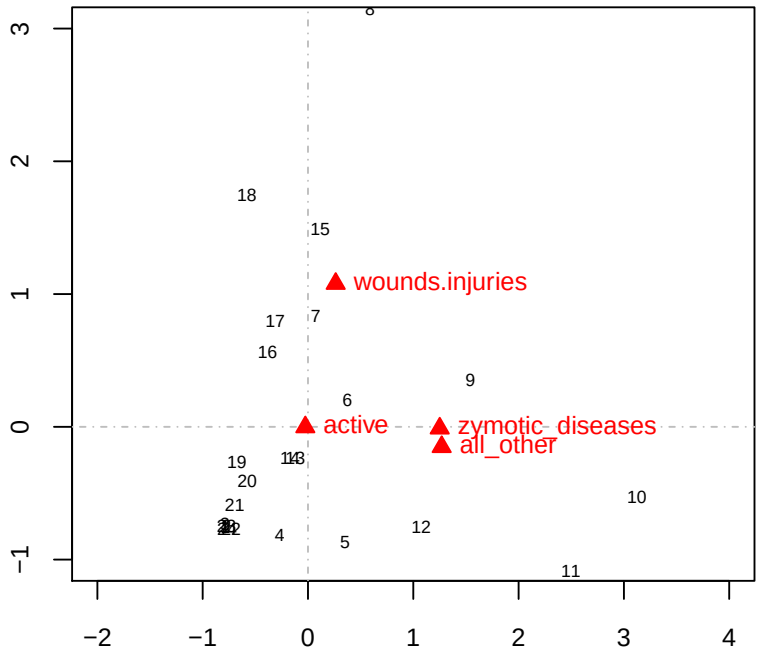
El gráfico asimétrico:

```
> plot(ntg.ca, map="colprincipal")
```



Mejor si etiquetamos los meses (filas):

```
> col.cp <- ntg.ca$colcoord %*% diag(ntg.ca$sv)
> row.ec <- ntg.ca$rowcoord
> par(cex.main=1,cex.lab=0.8,cex.axis=0.8)
> plot(col.cp[,1:2],type="n",ylim=c(-1,3),xlim=c(-2,4),xlab="",ylab="")
> text(row.ec[,1],row.ec[,2],1:24,cex=0.6)
> abline(h=0,v=0,lty=4,col="gray")
> points(col.cp[,1],col.cp[,2], pch=24,col="red",bg="red")
> text(col.cp[,1],col.cp[,2],labels=colnames(ntg[,5:8]),pos=4,cex=0.8,col="red")
```



Se observa que el eje principal con una inercia del 91 % separa los activos (a la izquierda) de los heridos o enfermos a la derecha. El segundo eje, con una importancia del 7.28 %, se caracteriza por los heridos (*Wounds & injuries*).
En cuanto a los primeros 12 meses consecutivos, aunque no todos, está claro que se situan hacia la derecha, con mayor abundancia de "no activos". Los últimos meses (19-24) están hacia la izquierda y muestran la mayor abundancia de soldados "activos". Las propuestas sanitarias de la enfermera Florence Nightingale mejoran la situación de los heridos y enfermos.