

# Modelos para crear mapas de posiconamiento

*Jordi López Sintas*

*13 de enero de 2015*

## Introducción

Supongamos que una muestra de consumidores ha valorado su percepción acerca de  $p$  atributos o características de un conjunto de  $n$  marcas. Cuando el número de atributos o características es elevado, es difícil hacerse una idea de cuán parecidas o diferentes son las marcas según la percepción que de ellas tienen los consumidores. Por ello nos preguntamos, ¿es posible representar visualmente  $n$  marcas en un espacio de reducidas dimensiones, por ejemplo,  $k$  dimensiones, siendo  $k < p$ ? ¿Y, además, que las marcas similares (según la percepción de los consumidores) se sitúen cerca unas de otras? Para ello disponemos de una familia de modelos conocida con el nombre de análisis o escalado multidimensional (MDS, multidimensional scaling). Esta familia de modelos, también, nos proporciona una medida de la discrepancia entre las percepciones observadas y las estimadas por el modelo, un índice de la medida en la que distancias observadas difieren de las estimadas.

Cuando las medidas acerca de las percepciones se han recogido en una escala métrica o intervalo el análisis multidimensional utiliza el análisis de los componentes principales para su representación visual (Gower y Hand, 1996). En general buscaremos una representación bidimensional, o tridimensional si la primera no reproduce las percepciones de una manera aceptable. Para ello partiremos de una matriz de percepciones,  $\mathbf{X}$ , que contendrá tantas filas,  $n$ , como marcas y tantas columnas,  $p$ , como atributos o características describan a los productos. El subespacio de  $k$  dimensiones proporcionado por los componentes principales pasa por el centro de los datos, por esa razón tenemos que transformar la matriz de datos original,  $\mathbf{X}$ , sustrayéndole la media de cada una de las medidas. Es habitual, asimismo, dividir las observaciones por su desviación típica, resultando una matriz de datos estandarizados.

La descomposición de la matriz de covarianzas, o correlaciones en la mayoría de los casos,  $X'X$ , nos proporciona los vectores y valores propios necesarios para construir e interpretar el espacio reducido,  $X'X = \Lambda\Lambda'$ , donde  $\Lambda$  es una matriz ortogonal que proporciona los vectores propios,  $\Lambda'\Lambda = I$ . Además, los valores de la diagonal de la matriz  $\Lambda$  nos proporcionan la varianza explicada por cada uno de los componentes principales (Gower y Hand, 1996). Los vectores propios nos proporcionan un espacio de  $k < p$  dimensiones alternativo al espacio de las características o atributos en los que se midieron las marcas que estamos analizando. Pero, también, podemos describir el significado de los nuevos componentes o beneficios que estructuran el mercado (o, alternativamente, de los vectores propios) analizando la correlación de los atributos con los componentes o factores del nuevo espacio.

El mejor espacio reducido para representar al conjunto de marcas viene dado por los ejes formados por los  $k$  primeros componentes principales estimados. Buscamos una combinación lineal de las variables originales como la siguiente:

$$C_1 = a_{11}x_1 + \dots + a_{1p}x_p$$

$$C_2 = a_{21}x_1 + \dots + a_{2p}x_p$$

...

$$C_k = a_{k1}x_1 + \dots + a_{kp}x_p$$

Con ciertas propiedades:

1. Que la varianza de  $C_i$  sea tan grande como sea posible
2. Que los valores de  $C_1, \dots, C_p$  para todos los individuos de la muestra sean independientes,  $S_{ij} = 0$ , para todos  $i$  y cada uno de los componentes

3. Para cada componente la suma del cuadrado de los coeficientes de su combinación lineal sea igual a la unidad,  $a_{11}^2 + a_{12}^2 + \dots + a_{1p}^2 = 1$ .

De hecho la proyección de las marcas en el espacio reducido a  $k$  ejes principales, vendrá dada por predicción del conjunto inicial de medidas en el nuevo subespacio definido por los  $k < p$  primeros componentes principales  $C = XA_k$ , por ejemplo  $k=2$ .

$$\begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ \vdots & \vdots \\ c_{n1} & c_{n2} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ \vdots & \vdots \\ a_{n1} & a_{n2} \end{pmatrix}$$

Así la predicción para la primera marca en el primer componente principal (primer elemento de la primera columna de la matriz  $C$ ) vendrá dada por

$$C_{11} = a_{11}x_{11} + a_{21}x_{21} + \dots + a_{p1}x_{p1}$$

y así sucesivamente con el resto de los factores o beneficios que estructuran el mercado.

¿Cuál es la calidad de la representación visual? Nuestro propósito es encontrar un subespacio  $k$  que minimice la diferencia entre las percepciones medidas y las que visualizamos en el mapa de percepciones. Para conocer la calidad de la representación visual tendremos, primero, que estimar las percepciones que se derivan del modelo visual,  $\hat{X} = C_{k=2}A'_{k=2}$ ,

$$\begin{pmatrix} \hat{x}_{11} & \hat{x}_{12} & \cdots & \hat{x}_{1p} \\ \hat{x}_{21} & \hat{x}_{22} & \cdots & \hat{x}_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{x}_{n1} & \hat{x}_{n2} & \cdots & \hat{x}_{np} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \\ \vdots & \vdots \\ c_{n1} & c_{n2} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{21} & \cdots & a_{p1} \\ a_{12} & a_{22} & \cdots & a_{p2} \end{pmatrix}$$

y, después, verificar la diferencia con los datos originales,  $X - \hat{X}$ . La suma cuadrado de las diferencias entre lo observado y lo predicho por el modelo,  $\text{traza}(X - \hat{X})'(X - \hat{X})$  resulta que es idéntica a la varianza no explicada por los componentes principales utilizados para formar el mapa de percepciones,  $\text{traza}(\Lambda - \Lambda_k)$ . Por ello, cuanto menor sea esa diferencia, mejor será la representación visual de las marcas en el espacio reducido mostrado en el mapa de percepciones. Si lo expresamos en relación a la variación total de la muestra, la  $\text{traza}(\Lambda)$ , la calidad de la representación tomará valores entre cero y uno, de tal manera que cuanto más se acerque a cero, mejor será la representación.

También es de interés conocer la calidad de la representación las variables originales en los componentes principales. Una medida de esa calidad nos la da la suma del cuadrado de los coeficientes de las funciones de los componentes principales,  $A_k A'_k 1$ , cuyo resultado es un vector columna con tantas filas como variables originales –donde 1 es el vector identidad. Si una variable está perfectamente representada, su valor del producto anterior será la unidad. Otra medida ampliamente utilizada es el coeficiente de correlación entre las variables originales y los componentes principales, que cuando ambas variables están estandarizadas es igual al coeficiente de la función principal multiplicado por la desviación estándar del componente,  $r_{ik} = a_{ik} \sqrt{\lambda_k} = l_{ik}$  (en la literatura anglosajona recibe el nombre de loadings), y su cuadrado nos da la medida en la que el componente  $k$  contribuye a explicar la varianza de la variable  $i$ . La calidad de representación visual de una variable original, entonces, será proporcional a la varianza total explicada por los componentes utilizados en la formación del mapa de percepciones,  $h_i^2 = \sum_k r_{ik}^2 = \sum_k a_{ik}^2 \lambda_k$ .

En ocasiones es necesario rotar los componentes sobre su propio eje con el objeto de facilitar la interpretación del significado de los componentes o beneficios que estructuran el mercado. En tales casos procedemos, primero, a normalizar los componentes, que entonces reciben el nombre de factores principales,  $\hat{X} = C_k \sum_k m1 \sum_k A'_k = F_k L_k$ , y  $F_j = \frac{C_j}{\sqrt{\text{Var}(C_j)}}$ . Así la relación entre las variables originales y los, ahora, factores principales, vendrá dada por  $\hat{X} = F_k L_k$  (Afifi y Azen, 1979).

$$\begin{pmatrix} \hat{x}_{11} & \hat{x}_{12} & \cdots & \hat{x}_{1p} \\ \hat{x}_{21} & \hat{x}_{22} & \cdots & \hat{x}_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{x}_{n1} & \hat{x}_{n2} & \cdots & \hat{x}_{np} \end{pmatrix} = \begin{pmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \\ \vdots & \vdots \\ f_{n1} & f_{n2} \end{pmatrix} = \begin{pmatrix} l_{11} & l_{21} & \cdots & l_{p1} \\ l_{12} & l_{22} & \cdots & l_{p2} \end{pmatrix}$$

Donde  $l_{ik} = a_{ik}\sqrt{\lambda_k}$  y a la matriz  $L_k$  se la conoce con el nombre de *matriz de correlaciones* (*pattern matrix*), a las cuales también se las denomina cargas (*loadings*). Ahora la rotación de factores principales (la puntuación de las marcas en ellos) no altera su propiedad de independencia entre los factores principales y corresponden a rotaciones ortogonales de las correlaciones (Venables y Ripley, 2000: 47-48). Una vez realizada la rotación, la nueva matriz de correlaciones,  $L^0$ , recibe el nombre de *matriz de la estructura* de los factores principales y trata de mostrar una estructura simple, fácilmente interpretable. Si la matriz de correlaciones no se ha rotado, coincide con la matriz de estructura de los factores.

Veamos cómo representar visualmente las percepciones que una muestra de consumidores mostraron acerca de un conjunto de marcas de coches.

```
g20per<-read.table("g20per.txt", header=TRUE)
names(g20per)
```

```
## [1] "G20"      "FORD"      "AUDI"      "TOYOTA"    "EAGLE"     "HONDA"     "SAAB"
## [8] "PONTIAC"  "BMW"       "MERCURY"
```

Para hacer el análisis tenemos que transponer la tabla de datos que acabamos de leer, de tal manera que las filas ahora serán las marcas, y las columnas, los atributos.

```
## [1] "Atract"    "Silec"      "Nofiab"     "MalaCons"   "Interesan"
## [6] "Deport"    "Incomod"    "Espacioso" "FacilMan"   "Prestigio"
## [11] "Ordinar"   "Economico" "Existos"    "Vanguard"   "Malacompra"
```

```
##           Atract Silec Nofiab MalaCons Interesan Deport Incomod Espacioso
## Atract      1.00  0.79 -0.87   -0.95      -0.6   -0.25  -0.570    0.36
## Silec        0.79  1.00 -0.52   -0.87      -0.7   -0.63  -0.755    0.53
## Nofiab       -0.87 -0.52  1.00    0.84       0.4   -0.09   0.214    0.03
## MalaCons     -0.95 -0.87  0.84    1.00       0.6    0.39   0.639   -0.41
## Interesan    -0.64 -0.70  0.37    0.60       1.0    0.54   0.542   -0.48
## Deport       -0.25 -0.63 -0.09    0.39       0.5    1.00   0.671   -0.74
## Incomod      -0.57 -0.76  0.21    0.64       0.5    0.67   1.000   -0.90
## Espacioso    0.36  0.53  0.03   -0.41      -0.5   -0.74  -0.904    1.00
## FacilMan     -0.17 -0.31 -0.06    0.29       0.4    0.74   0.720   -0.85
## Prestigio     0.87  0.75 -0.76   -0.92      -0.5   -0.46  -0.711    0.57
## Ordinar      -0.68 -0.42  0.46    0.60       0.5    0.32   0.640   -0.72
## Economico    -0.06  0.09  0.25    0.16      -0.6   -0.22   0.003    0.07
## Existos       0.93  0.74 -0.84   -0.90      -0.5   -0.24  -0.602    0.39
## Vanguard     0.18 -0.15 -0.18   -0.03      -0.1    0.48   0.114   -0.11
## Malacompra   -0.67 -0.54  0.43    0.49       0.8    0.08   0.342   -0.22
##           FacilMan Prestigio Ordinar Economico Existos Vanguard
## Atract       -0.17      0.9   -0.7   -0.055    0.93   0.177
## Silec         -0.31      0.7   -0.4    0.086    0.74  -0.151
## Nofiab        -0.06     -0.8    0.5    0.252   -0.84  -0.185
## MalaCons      0.29     -0.9    0.6    0.164   -0.90  -0.033
## Interesan     0.36     -0.5    0.5   -0.645   -0.52  -0.131
## Deport        0.74     -0.5    0.3   -0.217   -0.24   0.475
```

```
## Incomod      0.72      -0.7      0.6      0.003     -0.60      0.114
## Espacioso  -0.85       0.6     -0.7      0.070      0.39     -0.105
## FacilMan     1.00     -0.5      0.5      0.022     -0.22      0.176
## Prestigio   -0.51      1.0     -0.7     -0.240      0.93     -0.113
## Ordinar      0.52     -0.7      1.0      0.099     -0.63     -0.258
## Economico    0.02     -0.2      0.1      1.000     -0.12     -0.007
## Existos     -0.22      0.9     -0.6     -0.121      1.00     -0.058
## Vanguard     0.18     -0.1     -0.3     -0.007     -0.06      1.000
## Malacompra  -0.08     -0.4      0.4     -0.589     -0.58     -0.340
##
## Malacompra
## Atract      -0.67
## Silec       -0.54
## Nofiab       0.43
## MalaCons     0.49
## Interesan    0.79
## Deport       0.08
## Incomod      0.34
## Espacioso  -0.22
## FacilMan    -0.08
## Prestigio   -0.37
## Ordinar      0.41
## Economico   -0.59
## Existos     -0.58
## Vanguard    -0.34
## Malacompra   1.00
```

Podemos hacer la descomposición de la matrix de correlaciones de la siguiente manera:

```
#centrar y dividir por la desviación estándar
X<-scale(g20)
X
```

```
##          Atract Silec Nofiab MalaCons Interesan Deport Incomod Espacioso
## G20        1.1   1.7  -0.7   -1.3   -0.60 -0.95  -0.48   0.07
## FORD       -1.0  -0.7   1.0    1.0    1.40  0.09   0.68  -0.30
## AUDI       -0.2   0.7   0.4   -0.4   -0.03 -1.35  -1.63   1.43
## TOYOTA      1.1  -0.2  -1.8   -0.9    0.40  1.79   0.25  -0.79
## EAGLE      -1.0  -0.8   1.1    1.1   -0.17  0.09   0.68  -0.66
## HONDA       0.5   0.9  -0.3   -0.3   -0.88  0.35  -0.33  -0.30
## SAAB        0.7   0.4   0.4   -0.3   -0.88 -0.69  -1.49   2.04
## PONTIAC     -1.1  -1.5   0.6    1.2    1.97  1.14   1.11  -1.03
## BMW         1.2   0.5  -1.4   -1.2   -1.03 -0.95  -0.04   0.20
## MERCURY     -1.1  -1.0   0.7    1.1   -0.17  0.48   1.26  -0.66
##
##          FacilMan Prestigio Ordinar Economico Existos Vanguard Malacompra
## G20         0.3     0.65   0.07   -0.6     0.5     0.3    -0.3
## FORD         0.9    -0.94   0.20   -0.5    -0.9    -1.2     1.2
## AUDI        -1.9     0.82  -0.07   -0.6     0.2    -1.2     1.2
## TOYOTA       0.9     0.57  -0.74   -1.2     0.8     1.6    -0.2
## EAGLE        0.3    -1.53   1.14    1.3    -1.5     0.5     0.0
## HONDA        1.1     0.07   0.60    1.4     0.9    -0.6    -1.7
## SAAB        -1.3     0.65  -2.08    0.4     0.5     1.1    -1.2
## PONTIAC       0.5    -0.69   1.14   -1.4    -0.6    -0.2     1.2
## BMW         -0.7     1.49  -0.87    0.4     1.3    -1.0    -0.5
## MERCURY       0.3    -1.11   0.60    0.8    -1.3     0.7     0.3
```

```
## attr("scaled:center")
##      Atract      Silec      Nofiab      MalaCons      Interesan      Deport
##          5          4          3          3          4          5
##      Incomod  Espacioso  FacilMan  Prestigio      Ordinar  Economico
##          4          4          4          5          3          4
##      Existos  Vanguard  Malacompra
##          5          4          4
## attr("scaled:scale")
##      Atract      Silec      Nofiab      MalaCons      Interesan      Deport
##          0.8      1.1      0.8      1.1      0.7      0.8
##      Incomod  Espacioso  FacilMan  Prestigio      Ordinar  Economico
##          0.7      0.8      0.5      1.2      0.7      0.7
##      Existos  Vanguard  Malacompra
##          0.8      0.5      0.6

##matriz de correlaciones, X'X
cor=X%*%t(X)
cor
```

```
##      G20 FORD AUDI TOYOTA EAGLE HONDA  SAAB PONTIAC BMW MERCURY
## G20      9  -7   3   3.3 -7.3   3.3   4.2  -8.7   7   -8
## FORD     -7  12  -1  -4.6  6.2  -4.9  -9.4  11.9  -9   6
## AUDI      3  -1  15  -6.9 -6.1  -3.9   6.9  -5.1   5  -7
## TOYOTA    3  -5  -7  15.4 -6.5   0.4  -0.6   0.7   2  -3
## EAGLE    -7   6  -6  -6.5 12.9  -0.8  -6.6   7.1 -10  11
## HONDA     3  -5  -4   0.4 -0.8   9.9   0.9  -7.1   4  -2
## SAAB      4  -9   7  -0.6 -6.6   0.9  18.1 -13.5   7  -7
## PONTIAC  -9  12  -5   0.7  7.1  -7.1 -13.5  18.3 -12   8
## BMW       7  -9   5   2.4 -10.0  4.4   6.9 -11.8  14  -9
## MERCURY  -8   6  -7  -3.5 11.1  -2.3  -6.8   8.3  -9  11
```

```
#Calcular los componentes principales P or t(P)
E=eigen(cor,TRUE)
E
```

```
## $values
## [1] 7e+01 3e+01 2e+01 1e+01 4e+00 3e+00 2e+00 6e-01 2e-01 7e-15
##
## $vectors
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## [1,] 0.28 -0.12 0.07 0.24 0.72 -0.09 -0.37 -0.28 0.075 -0.3
## [2,] -0.34 0.17 0.23 0.17 -0.20 0.38 -0.62 0.32 -0.048 -0.3
## [3,] 0.21 0.59 0.37 0.11 0.16 -0.07 0.45 0.35 0.077 -0.3
## [4,] 0.09 -0.68 0.30 -0.35 0.04 -0.04 0.13 0.44 -0.107 -0.3
## [5,] -0.34 0.13 -0.45 -0.04 0.20 -0.24 0.10 0.08 -0.669 -0.3
## [6,] 0.14 -0.23 -0.46 0.43 -0.09 0.53 0.34 0.05 0.164 -0.3
## [7,] 0.37 0.26 -0.19 -0.69 -0.12 0.30 -0.13 -0.24 -0.014 -0.3
## [8,] -0.46 -0.08 0.41 -0.02 -0.08 0.12 0.31 -0.63 -0.009 -0.3
## [9,] 0.38 -0.06 0.04 0.31 -0.59 -0.47 -0.14 -0.18 -0.158 -0.3
## [10,] -0.34 0.02 -0.30 -0.16 -0.03 -0.42 -0.05 0.10 0.689 -0.3
```

```
A<-E$variables
```

```
A
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## [1,]  0.28 -0.12  0.07  0.24  0.72 -0.09 -0.37 -0.28  0.075 -0.3
## [2,] -0.34  0.17  0.23  0.17 -0.20  0.38 -0.62  0.32 -0.048 -0.3
## [3,]  0.21  0.59  0.37  0.11  0.16 -0.07  0.45  0.35  0.077 -0.3
## [4,]  0.09 -0.68  0.30 -0.35  0.04 -0.04  0.13  0.44 -0.107 -0.3
## [5,] -0.34  0.13 -0.45 -0.04  0.20 -0.24  0.10  0.08 -0.669 -0.3
## [6,]  0.14 -0.23 -0.46  0.43 -0.09  0.53  0.34  0.05  0.164 -0.3
## [7,]  0.37  0.26 -0.19 -0.69 -0.12  0.30 -0.13 -0.24 -0.014 -0.3
## [8,] -0.46 -0.08  0.41 -0.02 -0.08  0.12  0.31 -0.63 -0.009 -0.3
## [9,]  0.38 -0.06  0.04  0.31 -0.59 -0.47 -0.14 -0.18 -0.158 -0.3
## [10,] -0.34  0.02 -0.30 -0.16 -0.03 -0.42 -0.05  0.10  0.689 -0.3
```

```
I=A%*%t(A)
```

```
I
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
## [1,] 1e+00 -1e-16 1e-17 -4e-17 1e-16 -2e-16 1e-16 6e-17 6e-17
## [2,] -1e-16 1e+00 -4e-17 2e-16 -2e-16 -1e-17 -1e-16 2e-16 1e-16
## [3,] 1e-17 -4e-17 1e+00 -3e-16 1e-16 2e-16 -2e-16 -5e-16 1e-17
## [4,] -4e-17 2e-16 -3e-16 1e+00 -3e-16 -3e-16 2e-16 1e-16 4e-16
## [5,] 1e-16 -2e-16 1e-16 -3e-16 1e+00 0e+00 -2e-16 -2e-16 -2e-16
## [6,] -2e-16 -1e-17 2e-16 -3e-16 0e+00 1e+00 1e-16 -8e-17 1e-16
## [7,] 1e-16 -1e-16 -2e-16 2e-16 -2e-16 1e-16 1e+00 0e+00 8e-17
## [8,] 6e-17 2e-16 -5e-16 1e-16 -2e-16 -8e-17 0e+00 1e+00 6e-17
## [9,] 6e-17 1e-16 1e-17 4e-16 -2e-16 1e-16 8e-17 6e-17 1e+00
## [10,] 2e-16 1e-16 -1e-16 -8e-17 1e-16 6e-17 -1e-16 -1e-17 -1e-16
##      [,10]
## [1,] 2e-16
## [2,] 1e-16
## [3,] -1e-16
## [4,] -8e-17
## [5,] 1e-16
## [6,] 6e-17
## [7,] -1e-16
## [8,] -1e-17
## [9,] -1e-16
## [10,] 1e+00
```

```
var<-E$values
```

```
#La puntuación en los componentes o factors principales
```

```
C = A %*% X
```

```
options(digits=6)
```

```
C
```

```
##      Atract      Silec      Nofiab      MalaCons      Interesan      Deport
## [1,]  0.419425  0.552656 -0.532832 -0.6166355 -0.550406 -0.204307
## [2,] -0.477611 -0.472766 -0.350991  0.3111947  1.485809  1.149494
## [3,] -0.175937 -0.011206  0.590936  0.0907965  1.032715 -0.596207
## [4,]  0.105501  0.529842  0.279071 -0.3089183 -0.213327 -0.823491
```

```
## [5,] -1.219183 -1.496830 1.343217 1.6041014 1.373585 1.324799
## [6,] 2.030054 0.987529 -1.800185 -1.3860541 -0.980177 0.934299
## [7,] 0.255199 1.417845 0.554524 -0.3244169 -0.655785 -1.527554
## [8,] 0.898750 1.095885 -0.210615 -0.9092012 -1.406934 -1.208439
## [9,] 1.397542 1.178931 -1.534204 -1.7208816 0.319407 -0.205847
## [10,] 0.346928 -0.611034 -0.644419 -0.1778489 0.135626 -0.373089
##      Incomod   Espacioso   FacilMan   Prestigio   Ordinar   Economico
## [1,] 0.0727279 -0.7138948 0.355131 -0.1855666 0.755003 0.338828
## [2,] 0.5805758 -1.2732584 1.044550 -0.1230915 1.383127 -0.897948
## [3,] -0.8241710 0.9778762 -0.667709 0.2277211 -0.632887 -1.222950
## [4,] -1.1322427 0.8911443 -1.436189 0.6561658 0.255980 -0.352679
## [5,] 0.7704729 -0.5476922 0.976947 -1.7005067 0.431417 -0.198203
## [6,] -0.4433720 -0.1411699 1.011451 1.1203777 -1.081162 0.298986
## [7,] -0.4963841 0.3835393 0.391047 0.0550705 0.477509 0.875153
## [8,] -2.1665696 2.1038595 -1.662401 1.2075562 -1.621017 0.881193
## [9,] -0.8406219 0.4721556 -0.279788 1.5364012 -1.138158 -2.180697
## [10,] 0.5090558 -0.0513198 -0.549624 0.7065978 -0.738048 -0.179957
##      Existos   Vanguard   Malacompra
## [1,] -0.232188 0.2549587 -0.0994452
## [2,] 0.289459 -1.5627763 0.9538285
## [3,] -0.032040 -0.6633954 0.9028008
## [4,] 0.375085 -0.0377344 -0.0424135
## [5,] -1.409019 1.0256421 0.3563326
## [6,] 1.951860 1.1888271 -2.3650519
## [7,] 0.313017 -1.6234497 -0.6646833
## [8,] 1.058839 -0.4498159 -0.8759441
## [9,] 1.209995 0.3917333 0.5272252
## [10,] 0.492225 -0.7710896 0.2366049
```

```
#La desviación estandar de cada atributo de la matriz rotada
sdev = sqrt(diag((1/(dim(X)[2]-1)* A %%% cor %%% t(A))))
sdev
```

```
## [1] 0.461311 0.979078 0.711939 0.659651 1.191503 1.379996 0.843901
## [8] 1.340482 1.198730 0.509292
```

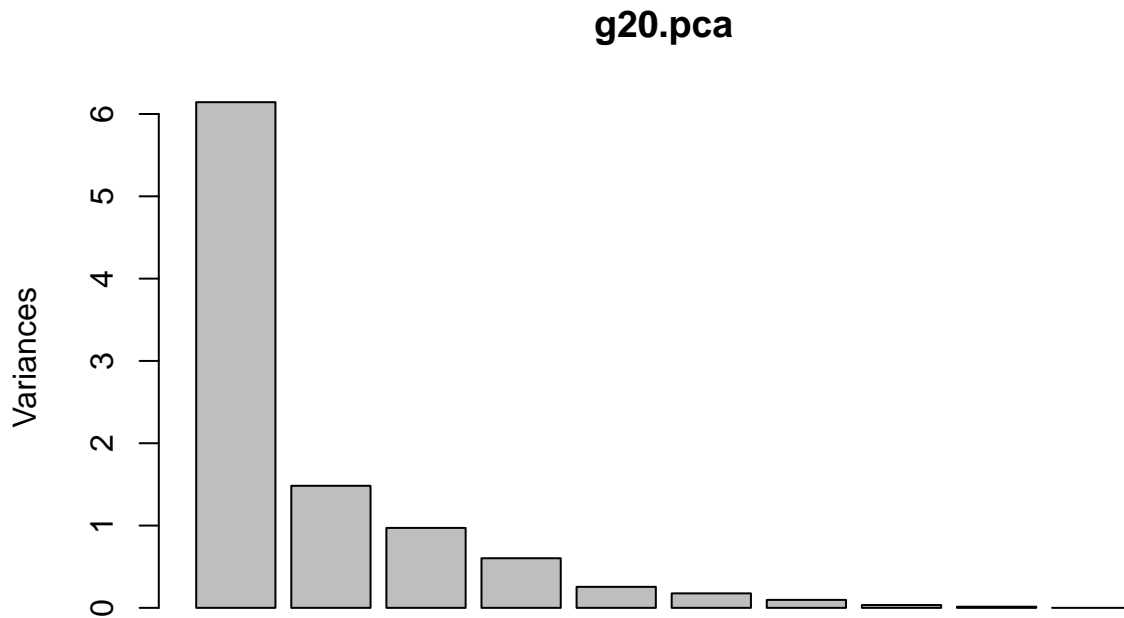
```
#####
```

No obstante podemos obtener los componentes principales utilizando las funciones `prcomp()` y `princomp()` de R. la equivalencia de resultados es la siguientes.

prcomp()	princomop()	Manualmente	Interpretación
sdev	sdev	sdev	Desviaciones estandr de cada columna de la matriz rotada
rotation	loadings	A	Los coeficients de los componentes principales
center	center	scale()	Dactos centrados en la media
scale	scale	scale()	Datos centrados y divididos por la desviación standar
X	scores	newdata	La puntuación en los componentes

Calculamos los componentes principales de la matriz de datos `g20` con la función `prcomp`. El resultado queda almacenado en el objeto `g20.pca`. La información está estructurada en diferentes componentes, `sdev` (las desviaciones estándares de los componentes principales), `rotation` (loadings), `center`, `scale` (la escala y centros utilizados), y `X` ( $\hat{X}$ ).

```
g20.pca<-prcomp(g20, cor=T)
plot(g20.pca)
```



```
summary(g20.pca)
```

```
## Importance of components:
##              PC1    PC2    PC3    PC4    PC5    PC6    PC7
## Standard deviation  2.479 1.218 0.9855 0.7763 0.5053 0.419 0.31145
## Proportion of Variance 0.628 0.152 0.0993 0.0616 0.0261 0.018 0.00992
## Cumulative Proportion 0.628 0.780 0.8794 0.9410 0.9672 0.985 0.99507
##              PC8    PC9    PC10
## Standard deviation  0.18515 0.11790 5.07e-16
## Proportion of Variance 0.00351 0.00142 0.00e+00
## Cumulative Proportion 0.99858 1.00000 1.00e+00
```

```
names(g20.pca)
```

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

```
g20.pca$sdev
```

```
## [1] 2.47863e+00 1.21774e+00 9.85529e-01 7.76323e-01 5.05308e-01
## [6] 4.19420e-01 3.11451e-01 1.85150e-01 1.17904e-01 5.07450e-16
```

```
g20.pca$rotation
```

```
##              PC1    PC2    PC3    PC4    PC5
## Atract      -0.29188890 0.1946836 -0.10726897 -0.0914134 0.0736472
## Silec       -0.39701528 -0.1231373 -0.22158169 0.4664042 0.4493927
## Nofiab       0.23129637 -0.4363021 0.01711390 0.0283078 0.2068075
```



```
## MalaCons      0.43071147 -0.2032741  0.00335342 -0.1636509 -0.1028254
## Interesan     0.19602465  0.1617329  0.43305407  0.1125605  0.1645701
## Deport        0.16464754  0.4629927 -0.04106408 -0.2905780  0.2311660
## Incomod       0.22009583  0.2713407 -0.11421903  0.0510549 -0.3091669
## Espacioso    -0.20346718 -0.4586310  0.20083258 -0.2749768  0.1458179
## FacilMan      0.09408365  0.2721296 -0.20710222  0.1092227  0.2062327
## Prestigio     -0.45725890  0.0981032  0.28267264 -0.0187148 -0.3802080
## Ordinar       0.21604211  0.0595714 -0.16582852  0.5781448 -0.0130735
## Economico     0.01213193 -0.2334333 -0.59086516 -0.1237918 -0.3521721
## Existos       -0.28404847  0.1823498  0.00461998 -0.0316698 -0.1616456
## Vanguard      0.00529025  0.1215812 -0.10439675 -0.3870939  0.4448913
## Malacompra    0.13161869 -0.0318702  0.43145462  0.2418743 -0.0810435
##               PC6      PC7      PC8      PC9      PC10
## Atract        0.0146376 -0.17552204 -0.1043880  0.4922212  0.5603096
## Silec         -0.0580889 -0.12098389  0.0265838 -0.3979857  0.1277295
## Nofiab        -0.1867461 -0.08615341 -0.3127262 -0.0532749  0.4159539
## MalaCons      -0.3081835 -0.00751391 -0.2104999 -0.2249212  0.1343760
## Interesan     -0.3086490 -0.17633906 -0.0834722  0.3076260 -0.2052955
## Deport        -0.3083159  0.35558389  0.3245623 -0.3133913  0.3551126
## Incomod       0.2735079 -0.39447763 -0.3516170 -0.3961055  0.0831602
## Espacioso    -0.1419905  0.01392458  0.0572559 -0.1560974 -0.2077360
## FacilMan      -0.2422852 -0.43703922  0.1213980 -0.1137630 -0.2924965
## Prestigio     -0.0808296  0.19351889 -0.2607870 -0.2838078  0.1209051
## Ordinar       -0.0569748  0.58298642 -0.2491678  0.1298428 -0.0841885
## Economico     -0.0952982  0.05960197  0.3001270  0.0569406 -0.0870917
## Existos       -0.4840675  0.04262098 -0.1975239 -0.0772476 -0.2466278
## Vanguard      0.4268468  0.24372281 -0.3509119 -0.0890720 -0.2684085
## Malacompra    0.2952736  0.02526712  0.4582625 -0.2104587  0.1123806
```

```
g20.pca$center
```

```
##      Atract      Silec      Nofiab      MalaCons      Interesan      Deport
##      4.78      4.41      3.42      3.09      4.02      4.83
##      Incomod      Espacioso      FacilMan      Prestigio      Ordinar      Economico
##      3.53      4.14      4.47      4.62      3.45      4.03
##      Existos      Vanguard      Malacompra
##      4.88      4.17      3.60
```

```
g20.pca$scale
```

```
## [1] FALSE
```

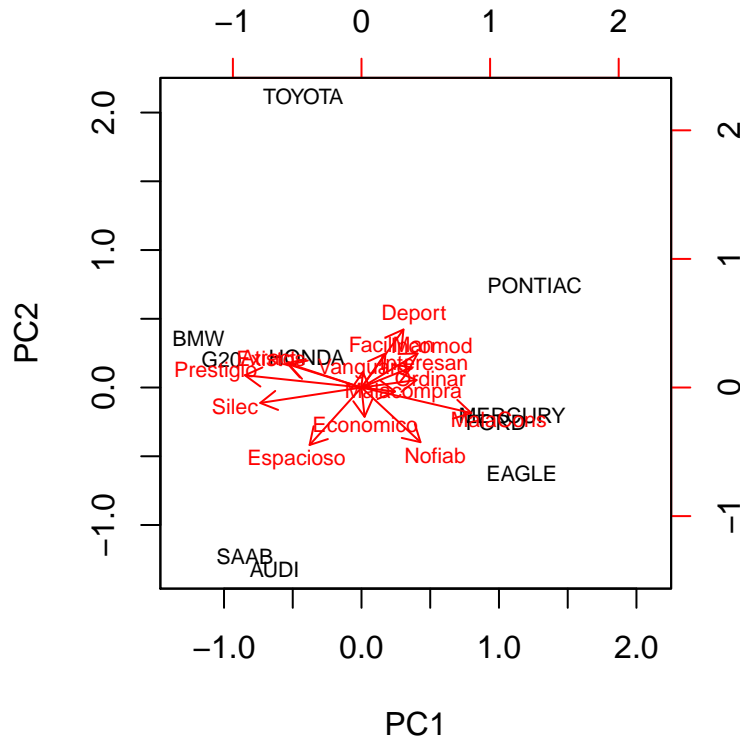
```
g20.pca$X
```

```
## NULL
```

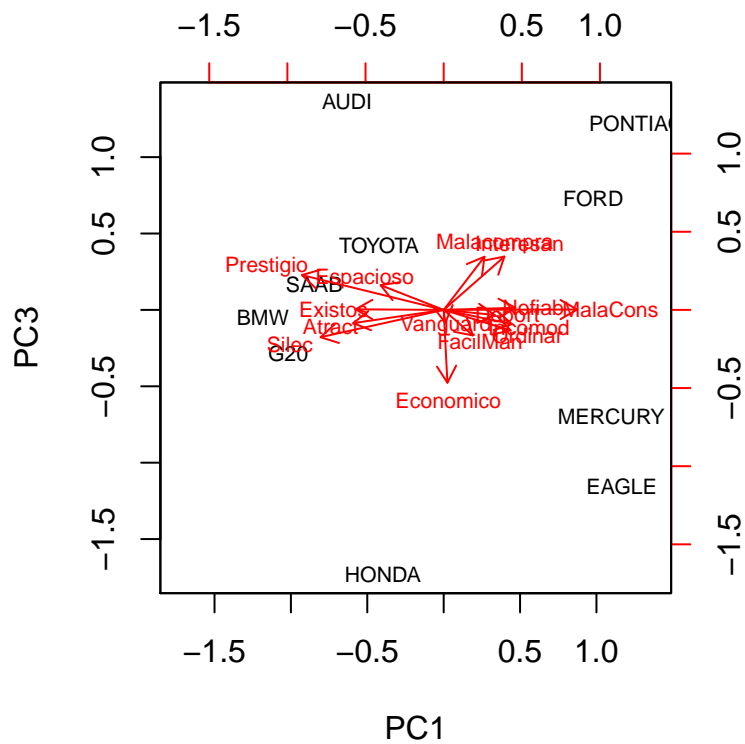
La representación simultánea de la puntuación de las marcas en el nuevo espacio reducido formado por los componentes principales y las variables originales se realiza por medio de la función `biplot` del paquete MASS. Las variables originales se representan por medio de unos vectores cuya longitud viene dada por su correlación con los factores principales, mayor cuanto mayor es la correlación. Nótese, no obstante, que la correlación de cada una de los atributos de las marcas con los factores y la puntuación de las marcas en ellos

forman dos grupos de datos medidos en diferentes escalas. Esa es la razón por la que en los gráficos aparecen dos escalas, una para cada grupo de datos. No obstante es habitual multiplicar las correlaciones por algún factor de escala como la varianza de los factores principales [insertar referencia a MDPREF]. De esa manera la proyección perpendicular de las marcas sobre los vectores de los atributos nos ofrece una medida relativa de la cantidad de ese atributo que los consumidores han percibido en la marca.

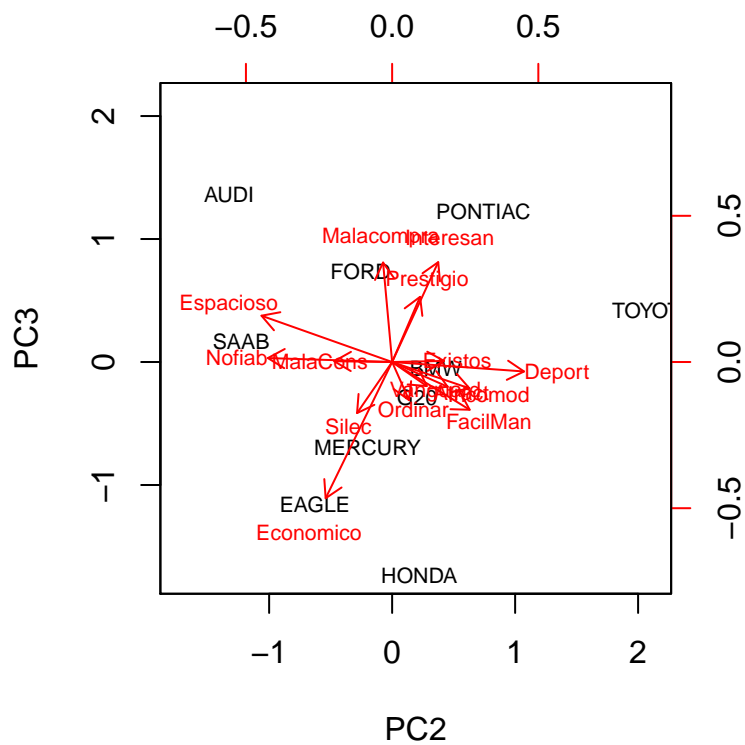
```
biplot(g20.pca, pc.biplot=T, cex=0.7, ex=0.8)
```



```
biplot(g20.pca, choices=c(1,3),pc.biplot=T, cex=0.7, ex=0.8)
```



```
biplot(g20.pca, choices=c(2,3),pc.biplot=T, cex=0.7, ex=0.8)
```



```
g20.puntos <- predict(g20.pca)
g20.puntos
```

##	PC1	PC2	PC3	PC4	PC5	PC6
----	-----	-----	-----	-----	-----	-----

```
## G20      -2.522887  0.247712 -0.281390  1.13690530  0.7154981  0.45115474
## FORD      2.426923 -0.312052  0.722705  0.32089204  0.0512457 -0.31173923
## AUDI     -1.565987 -1.613781  1.345426  0.67959981 -0.0231086  0.00509272
## TOYOTA   -1.047514  2.575236  0.416294 -0.74039098  0.3533305  0.15627428
## EAGLE     2.890005 -0.764345 -1.138404 -0.00727378  0.0579163  0.37853351
## HONDA    -0.985592  0.269983 -1.705520  0.40690256  0.0325804 -0.85362113
## SAAB     -2.099931 -1.494307  0.167256 -1.64017957  0.3523044 -0.16251301
## PONTIAC   3.122505  0.902919  1.205225  0.16018836 -0.0907612 -0.31709578
## BMW      -2.935520  0.431988 -0.046794  0.05015956 -1.2100478  0.17640370
## MERCURY   2.717998 -0.243354 -0.684799 -0.36680330 -0.2389578  0.47751020
##          PC7          PC8          PC9          PC10
## G20      -0.228911 -0.19087173 -0.02089742 -6.14092e-16
## FORD     -0.664410  0.20720894  0.00365967 -7.63278e-16
## AUDI      0.457441  0.19366408 -0.02912526 -4.85723e-16
## TOYOTA    0.148427  0.22449093  0.02242259 -1.02349e-16
## EAGLE     0.159360  0.05598570  0.23077960 -4.37150e-16
## HONDA     0.163989  0.03557697 -0.04754885 -5.55112e-16
## SAAB     -0.124490 -0.14072061  0.01192467 -6.10623e-16
## PONTIAC   0.209685 -0.33744921  0.02533885 -7.77156e-16
## BMW      -0.180563 -0.05582461  0.05614028 -9.02056e-17
## MERCURY   0.059473  0.00793955 -0.25269414 -3.43475e-16
```

## Mapas conjuntos: De percepciones y preferencias

### Análisis interno

Si queremos introducir las preferencias en el mapa y construir un mapa conjunto, de percepciones y preferencias, entonces necesitamos añadir las preferencias de los consumidores. Para ello debemos definir el modelo de preferencias, vectoriales o ideales. En el primer caso pedimos a los consumidores que nos digan su preferencias por las marcas, segmentamos la matriz de preferencias y el resultado lo incorporamos a la matriz de de percepciones como nuevos atributos que muestran las preferencias por las marcas. En el caso de utilizar un modelo ideal de preferencias, pediríamos a los consumidores que nos mostraran sus preferencias por cada uno de los atributos de forman las marcas, segmentaríamos e introduciríamos el resultado en la matriz de percepciones como nuevas marcas.

En el caso de las preferencias vectoriales

```
##analysis interno
#Leer fichero g20seg.txt
g20seg<-read.table("g20seg.txt", header=T)
head(g20seg)
```

```
##          G20 FORD AUDI TOYOTA EAGLE HONDA SAAB PONTIAC BMW MERCURY
## Atract    5.6  4.0  4.6    5.6  4.0   5.2  5.3    3.9 5.7    3.9
## Silec     6.3  3.6  5.2    4.2  3.5   5.4  4.8    2.8 5.0    3.3
## Nofiab    2.9  4.2  3.7    2.0  4.3   3.2  3.7    3.9 2.3    4.0
## MalaCons  1.6  4.2  2.6    2.1  4.3   2.8  2.8    4.4 1.8    4.3
## Interesan 3.6  5.0  4.0    4.3  3.9   3.4  3.4    5.4 3.3    3.9
## Deport    4.1  4.9  3.8    6.2  4.9   5.1  4.3    5.7 4.1    5.2
```

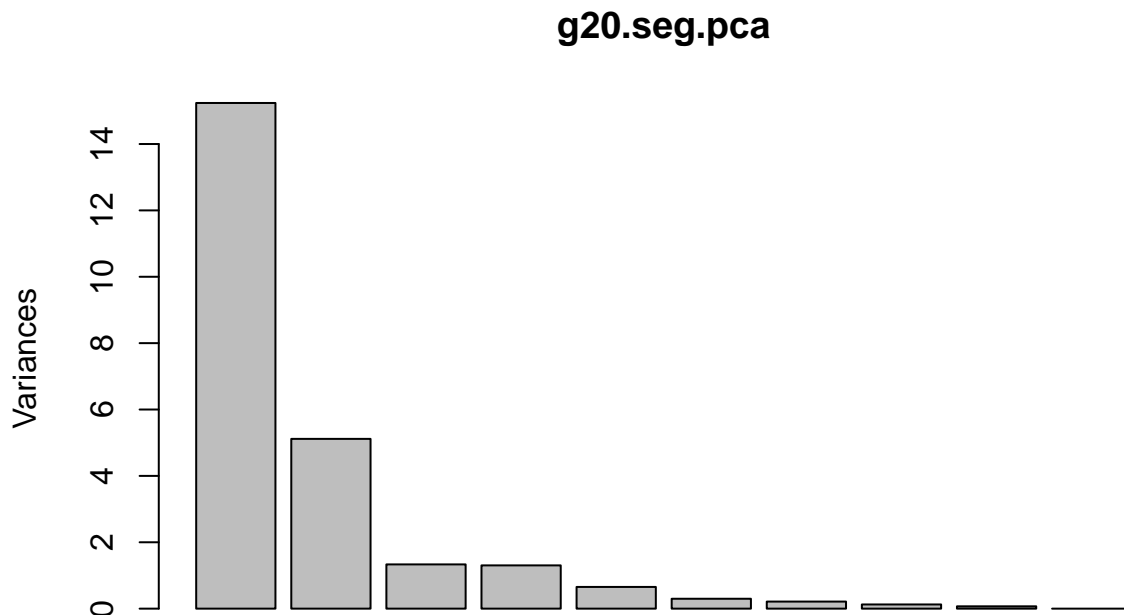
```
g20.seg<-data.frame(t(g20seg))
names(g20.seg)
```

```
## [1] "Atract"      "Silec"      "Nofiab"     "MalaCons"   "Interesan"
## [6] "Deport"      "Incomod"    "Espacioso" "FacilMan"   "Prestigio"
## [11] "Ordinar"     "Economico"  "Existos"    "Vanguard"   "Malacompra"
## [16] "S1"         "S2"         "S3"
```

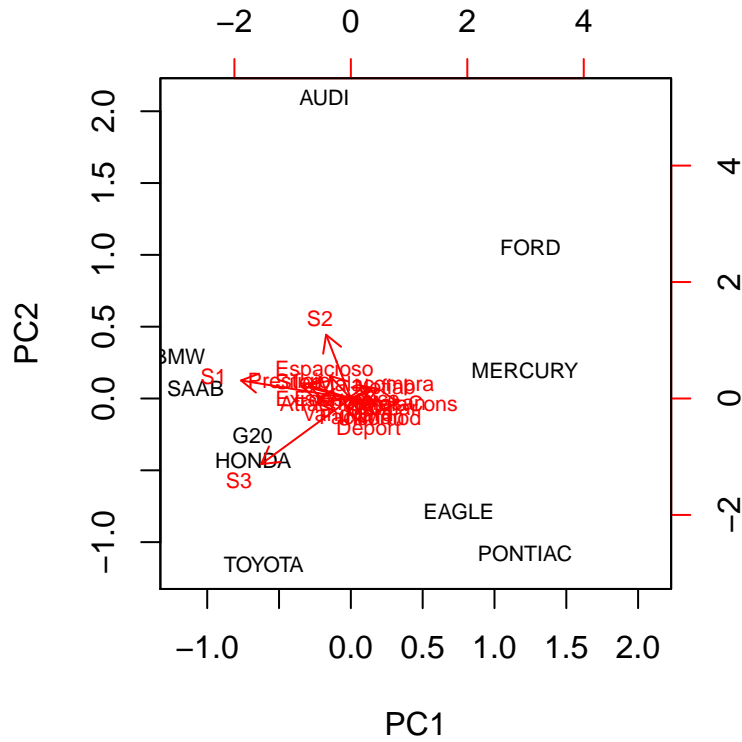
```
head(g20.seg)
```

```
##      Atract Silec Nofiab MalaCons Interesan Deport Incomod Espacioso
## G20      5.6  6.3   2.9     1.6      3.6    4.1    3.2     4.2
## FORD      4.0  3.6   4.2     4.2      5.0    4.9    4.0     3.9
## AUDI      4.6  5.2   3.7     2.6      4.0    3.8    2.4     5.3
## TOYOTA     5.6  4.2   2.0     2.1      4.3    6.2    3.7     3.5
## EAGLE      4.0  3.5   4.3     4.3      3.9    4.9    4.0     3.6
## HONDA      5.2  5.4   3.2     2.8      3.4    5.1    3.3     3.9
##      FacilMan Prestigio Ordinar Economico Existos Vanguard Malacompra
## G20          4.6      5.4     3.5      3.6     5.3     4.3     3.4
## FORD          4.9      3.5     3.6      3.7     4.2     3.6     4.3
## AUDI          3.5      5.6     3.4      3.6     5.0     3.6     4.3
## TOYOTA        4.9      5.3     2.9      3.2     5.5     4.9     3.5
## EAGLE          4.6      2.8     4.3      4.9     3.7     4.4     3.6
## HONDA          5.0      4.7     3.9      5.0     5.6     3.9     2.6
##      S1  S2  S3
## G20    4.3 5.9 8.6
## FORD    2.1 6.0 2.1
## AUDI    6.0 7.7 3.4
## TOYOTA  6.1 3.5 8.1
## EAGLE   3.3 3.1 5.8
## HONDA   6.0 5.5 8.3
```

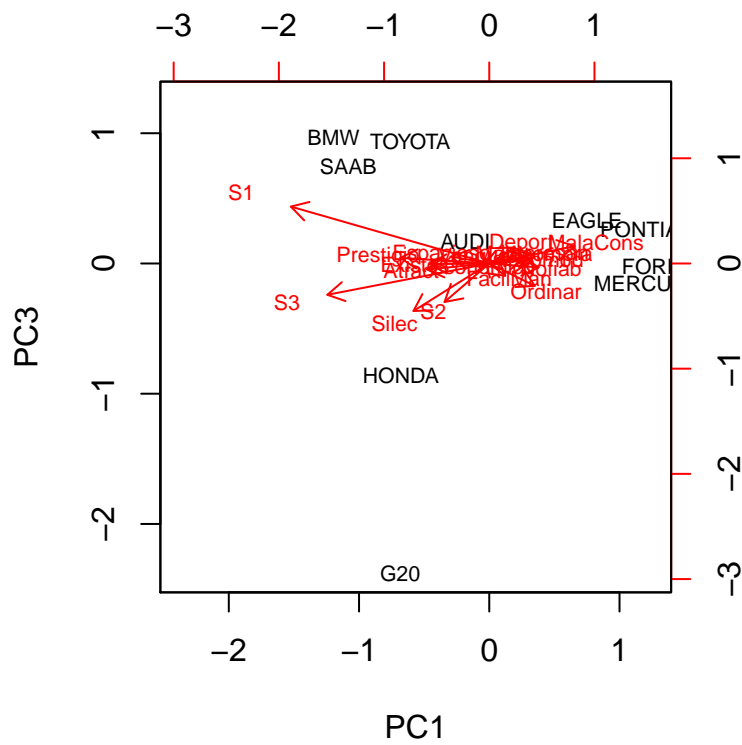
```
g20.seg.pca <- prcomp(g20.seg, cor=TRUE)
plot(g20.seg.pca)
```



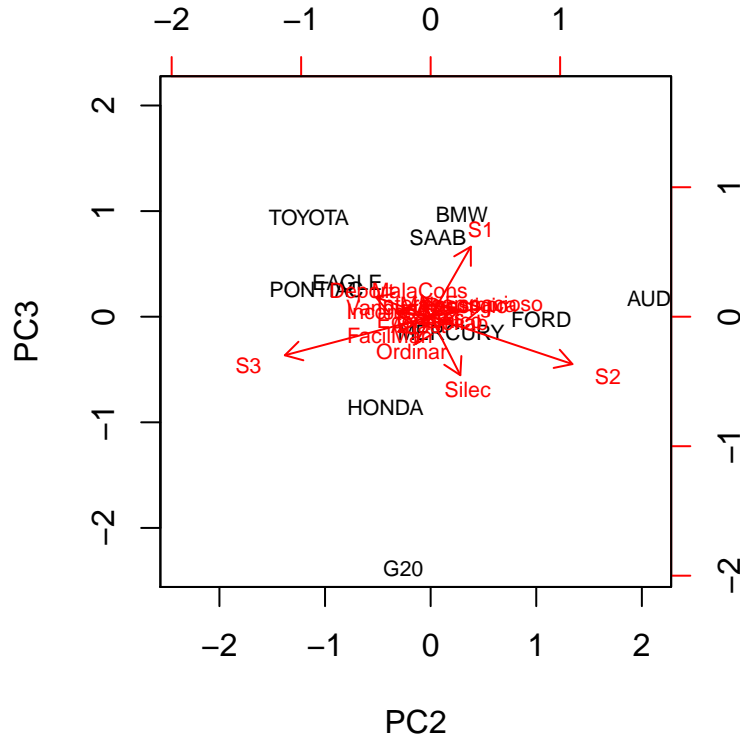
```
biplot(g20.seg.pca , pc.biplot=T, cex=0.7, ex=0.8)
```



```
biplot(g20.seg.pca , choices=c(1,3), pc.biplot=T, cex=0.7, ex=0.8)
```



```
biplot(g20.seg.pca , choices=c(2,3), pc.biplot=T, cex=0.7, ex=0.8)
```



## Análisis externo

Breakfast rating for rectangular SMACOF As a metric unfolding example we use the breakfast dataset from Green and Rao (1972) which is also analyzed in Borg and Groenen (2005, Chapter 14). 42 individuals were asked to order 15 breakfast items due to their preference. These items are: toast = toast pop-up, butoast = buttered toast, engmuff = English muffin and margarine, jdonut = jelly donut, cintoast = cinnamon toast, bluemuff = blueberry muffin and margarine, hrolls = hard rolls and butter, toastmarm = toast and marmalade, butoastj = buttered toast and jelly, toastmarg = toast and margarine, cinbun = cinnamon bun, danpastry = Danish pastry, gdonut = glazed donut, cofcake = coffee cake, and cornmuff = corn muffin and butter. For this  $42 \times 15$  matrix we compute a rectangular SMACOF solution.

The data set we provide for three-way SMACOF is described in Bro (1998). The raw data consist of ratings of 10 breads on 11 different attributes carried out by 8 raters. Note that the bread samples are pairwise replications: Each of the 5 different breads, which have a different salt content, was presented twice for rating. The attributes are bread odor, yeast odor, off-flavor, color, moisture, dough, salt taste, sweet taste, yeast taste, other taste, and total. First we fit an unconstrained solution followed by a model with identity restriction.