

# Escalado multidimensional métrico (EMM)

Edgar Ortiz Mota

2022-06-02

## Matriz de trabajo

En esta ocacion se va a trabajr con la matriz denominada **eurodist** precargada en R.

Cargamos la matriz de datos eurodist

```
data.dist<-eurodist  
data.dist
```

```
## Athens Barcelona Brussels Calais Cherbourg Cologne Copenhagen  
## Barcelona 3313  
## Brussels 2963 1318  
## Calais 3175 1326 204  
## Cherbourg 3339 1294 583 460  
## Cologne 2762 1498 206 409 785  
## Copenhagen 3276 2218 966 1136 1545 760  
## Geneva 2610 803 677 747 853 1662 1418  
## Gibraltar 4485 1172 2256 2224 2047 2436 3196  
## Hamburg 2977 2018 597 714 1115 460 460  
## Hook of Holland 3030 1490 172 330 731 269 269  
## Lisbon 4532 1305 2084 2052 1827 2290 2971  
## Lyons 2753 645 690 739 789 714 1458  
## Madrid 3949 636 1558 1550 1347 1764 2498  
## Marseilles 2865 521 1011 1059 1101 1035 1778  
## Milan 2282 1014 925 1077 1209 911 1537  
## Munich 2179 1365 747 977 1160 583 1104  
## Paris 3000 1033 285 280 340 465 1176  
## Rome 817 1460 1511 1662 1794 1497 2050  
## Stockholm 3927 2868 1616 1786 2196 1403 650  
## Vienna 1991 1802 1175 1381 1588 937 1455  
## Geneva Gibraltar Hamburg Hook of Holland Lisbon Lyons Madrid  
## Barcelona  
## Brussels  
## Calais  
## Cherbourg  
## Cologne  
## Copenhagen  
## Geneva  
## Gibraltar 1975  
## Hamburg 1118 2897  
## Hook of Holland 895 2428 550  
## Lisbon 1936 676 2671 2280  
## Lyons 158 1817 1159 863 1178
```

```
## Madrid          1439          698          2198          1730          668          1281
## Marseilles      425          1693          1479          1183          1762          320          1157
## Milan           328          2185          1238          1098          2250          328          1724
## Munich          591          2565          805           851          2507          724          2010
## Paris           513          1971          877           457          1799          471          1273
## Rome            995          2631          1751          1683          2700          1048          2097
## Stockholm       2068          3886          949           1500          3231          2108          3188
## Vienna          1019          2974          1155          1205          2937          1157          2409
##               Marseilles Milan Munich Paris Rome Stockholm
## Barcelona
## Brussels
## Calais
## Cherbourg
## Cologne
## Copenhagen
## Geneva
## Gibraltar
## Hamburg
## Hook of Holland
## Lisbon
## Lyons
## Madrid
## Marseilles
## Milan           618
## Munich          1109          331
## Paris           792          856          821
## Rome            1011          586          946          1476
## Stockholm       2428          2187          1754          1827          2707
## Vienna          1363          898          428          1249          1209          2105
```

Transformamos los datos en matriz

```
data.dist<-as.matrix(data.dist)
```

Exploracion de la matriz

```
dim(data.dist)
```

```
## [1] 21 21
```

```
###Extracción de las filas de la matriz
```

Numero de ciudades

```
n<-nrow(data.dist)
```

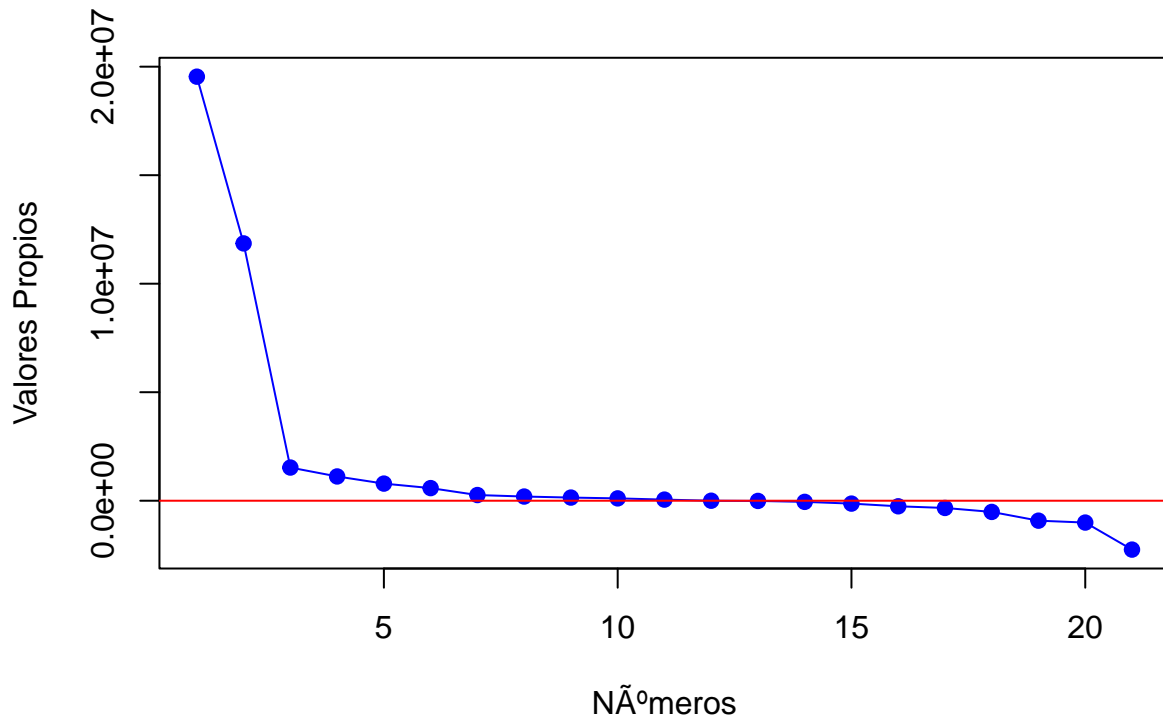
## Escalado multidimensional clásico

1.- cálculo de autovalores Dentro del objeto mds.cities se encuentran almacenado los valores propios (eigenvalues) en mds.cities\$eig

```
mds.cities<-cmdscale(data.dist, eig = TRUE)
```

2.- Generacion del grafico

```
plot(mds.cities$eig, pch=19, col="blue",
      xlab="Números", ylab="Valores Propios",
      type="o")
abline(a=0, b=0, col="red")
```



Interpretacion: se identifican autovalores negativos(son de la linea roja havia abajo) Se considera como solución el seleccionar r=2 coordenadas principales(las que estan mas altas).

3.- Medidas de precision

```
m<-sum(abs(mds.cities$eig[1:2]))/sum(abs(mds.cities$eig))
```

4.- Obtencion de coordenadas principales fijando k=2 y se realice con los dos primeros autovalores.

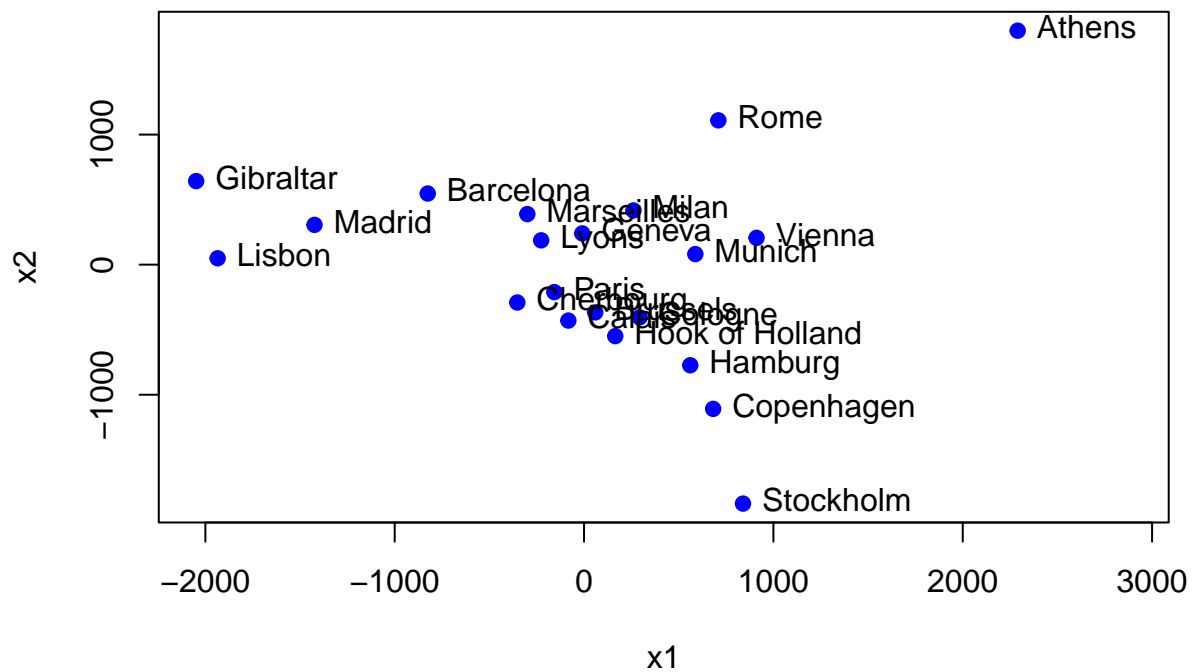
```
mds.cities<-cmdscale(data.dist, eig=TRUE, k=2)
```

```
x1<-mds.cities$points[,1]
```

```
x2<-mds.cities$points[,2]
```

5.- Generacion del gráfico en dos dimensiones de los datos con las coordenadas obtenidas(k=2)

```
plot(x1,x2,pch=19, col="blue",
     xlim = range(x1)+c(0,600))
text(x1,x2, pos=4, labels = rownames(data.dist),
     col="black")
```



Se invierten los ejes del plot(rotacion)

```
x2<--x2

plot2<-plot(x1,x2,pch=0, col="purple",
  xlim = range(x1)+c(0,600))
text(x1,x2, pos=4, labels = rownames(data.dist),
  col="black")
```

