**Ejercicio Propuesto 3 (Resuelto)**

**El fichero de datos**[***Localidades.txt***](https://wpd.ugr.es/~bioestad/wp-content/uploads/Localidades.txt)**incluye información sobre los servicios de los que disponen (señalados con un 1) y no disponen (señalados con un 0) una muestra de 10 localidades españolas de menos de 1.000 habitantes. Realizar un análisis clúster jerárquico aglomerativo utilizando el método del promedio entre grupos y considerando la distancia adecuada.**

**Solución**

En primer lugar, nos situamos en el directorio de trabajo

> setwd("C:/Users/Usuario/Desktop/Datos")

A continuación, importamos el fichero de texto utilizando la orden *read.table* y guardamos el resultado de la importación en una variable que, en nuestro caso, llamaremos *Localidades*. Como el fichero de texto contiene los nombres de las variables en la primera fila, asignamos el valor *TRUE* al parámetro *header*.

> Localidades <- read.table("Localidades.txt", header = TRUE)

Consultamos el contenido de la variable en la que hemos almacenado los datos, para asegurarnos de que la importación se ha realizado correctamente.

> Localidades

Localidad Consultorio Cine Teatro Parada.Bus Parada.Taxi Banco Biblioteca

1 1 1 0 1 1 0 1 0

2 2 0 1 0 1 1 0 0

3 3 0 0 0 1 0 1 1

4 4 1 0 1 1 1 1 0

5 5 0 1 0 1 1 0 1

6 6 1 0 1 0 0 1 0

7 7 1 1 0 1 0 0 0

8 8 0 0 0 1 1 0 1

9 9 1 1 0 1 0 1 1

10 10 0 0 0 1 1 0 1

Supermercado

1 1

2 0

3 1

4 0

5 1

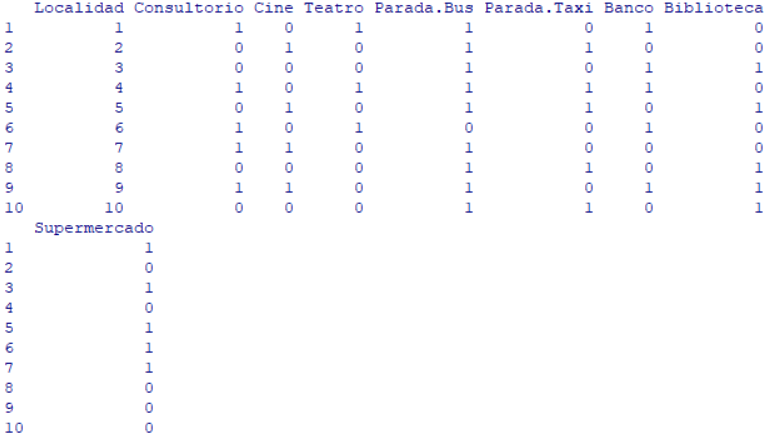
6 1

7 1

8 0

9 0

10 0



Como ya nos adelantaba el enunciado, todas las variables que componen el conjunto de datos son de tipo dicotómico y toman únicamente los valores 0 y 1. Consecuentemente, para calcular la distancia entre cada para de individuos (de localidades, en este caso) usaremos la distancia binaria. De forma previa al cálculo de estas distancias, crearemos un *data frame,* que llamaremos distancias, y contendrá únicamente las variables dicotómicas del conjunto de datos.

> variables <- Localidades[,-1]

> distancias <- dist(variables, method = "binary")

> distancias

1 2 3 4 5 6 7

2 0.8571429

3 0.5000000 0.8333333

4 0.3333333 0.6666667 0.7142857

5 0.7500000 0.4000000 0.5000000 0.7500000

6 0.2000000 1.0000000 0.6666667 0.5000000 0.8750000

7 0.5000000 0.6000000 0.6666667 0.7142857 0.5000000 0.6666667

8 0.8571429 0.5000000 0.6000000 0.6666667 0.4000000 1.0000000 0.8333333

9 0.5714286 0.6666667 0.5000000 0.5714286 0.5714286 0.7142857 0.5000000

10 0.8571429 0.5000000 0.6000000 0.6666667 0.4000000 1.0000000 0.8333333

8 9

2

3

4

5

6

7

8

9 0.6666667

10 0.0000000 0.6666667

A continuación, llamamos a la función *hclust*, indicando como argumentos la matriz de distancias que devuelve la función *dist* así como el método de aglomeración.

> cluster <- hclust(distancias, method = "average")  
> cluster

Call:  
hclust(d = distancias, method = "average")

Cluster method : average  
Distance : binary  
Number of objects: 10

Consultemos, en primer lugar, el historial de conglomeración del análisis, que está contenido en el elemento *merge* del objeto *cluster*.

> cluster$merge

[,1] [,2]

[1,] -8 -10

[2,] -1 -6

[3,] -2 -5

[4,] -4 2

[5,] 1 3

[6,] -3 -9

[7,] -7 6

[8,] 4 7

[9,] 5 8

Según lo mostrado en el historial de conglomeración, las localidades más similares en lo que a servicios se refiere son la 8 y la 10 y, en consecuencia, son las primeras que se unen en un grupo. A continuación, las localidades 1 y 6 hacen lo propio, pero en una agrupación independiente a la anterior. El resto de pasos del historial de conglomeración se interpreta del mismo modo.

Las distancias a las que se producen las fusiones entre localidades se pueden consultar accediendo al elemento *height* del objeto *cluster*.

> cluster$height

[1] 0.0000000 0.2000000 0.4000000 0.4166667 0.4500000 0.5000000 0.5833333

[8] 0.6243386 0.7424107

Así, se puede apreciar que la primera fusión entre localidades se produce a una distancia de 0 unidades; la segunda, a una distancia de 0.2 unidades; la tercera, a una distancia de 0.4 unidades y así, sucesivamente.

Por último, graficamos el dendrograma asociado al proceso de formación de clústers.

> plot(cluster)

