R-PL4

Gabriel López, Sergio Sanz, Álvaro Zamorano

12 de noviembre de 2019

1. Ejercicio realizado en clase.

A partir del siguiente conjunto de calificaciones académicas, pertenecientes a dos grupos de alumnos (mañana y tarde), formados por dos notas: teoría y laboratorio, las notas de teoría y laboratorio tendrán valores entre 0 y 5, realizar un análisis de clasificación no supervisada utilizando el algoritmo **K-Means**.

Alumno	Teoría	Laboratorio
A1	4	4
A2	3	5
A3	1	2
A4	5	5
A5	0	1
A6	2	2
A7	4	4
A8	2	1

En primer lugar se introducirán los datos en forma de matriz y se hará la traspuesta de esta.

```
> m<-matrix(c(4,4,3,5,1,2,5,5,0,1,2,2,4,5,2,1),2,8)
> (m<-t(m))

     [,1] [,2]
[1,]    4    4
[2,]    3    5
[3,]    1    2
[4,]    5    5
[5,]    0    1
[6,]    2    2
[7,]    4    5
[8,]    2    1
```

En primer lugar se deben seleccionar el número de clusters en los que se van a agrupar los datos, en este caso serán 2. Además es necesario indicar los

centroides iniciales de cada uno de ellos, en este caso son C1{0,1} y C2{2,2}. Todo ello es elegido de forma arbitraria.

Introducimos los centroides en una matriz y se realiza la traspuesta.

```
> c<-matrix(c(0,1,2,2),2,2)
> (c<-t(c))

     [,1] [,2]
[1,]    0    1
[2,]    2    2
```

La función `K-Means` se encuentra en el paquete **stats**. Dicho paquete se carga por defecto al arrancar R; para comprobarlo se hace uso de la función `search()`.

```
> search()

 [1] ".GlobalEnv"        "package:foreign"   "package:stats"
 [4] "package:graphics"  "package:grDevices" "package:utils"
 [7] "package:datasets"  "package:methods"   "Autoloads"
[10] "package:base"
```

Por último hacemos uso de la función y obtenemos los centroides finales. Indicamos que en número máximo de iteraciones es 4.

```
> (clasificacionns<-kmeans(m,c,4))

K-means clustering with 2 clusters of sizes 4, 4

Cluster means:
  [,1] [,2]
1 1.25 1.50
2 4.00 4.75

Clustering vector:
[1] 2 2 1 2 1 1 2 1

Within cluster sum of squares by cluster:
[1] 3.75 2.75
 (between_SS / total_SS =  84.8 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Los resultados obtenidos son los mismos que los de clase, es decir, **C1{1.25,1.5} y C2{4,4.75}.**

A continuación usaremos los clusters obtenios para separar los datos de la muestra en dos grupos. Para ello se hace uso de la función `cbind` la cuál añade (por delante) una columna a la matriz de datos. Dicha columna se corresponde con la clasificación obtenida, será 1 ó 2 dependiendo del cluster al que pertenezca cada muestra.

```
> (m = cbind(clasificacionns$cluster,m))
```

```
     [,1] [,2] [,3]
[1,]    2    4    4
[2,]    2    3    5
[3,]    1    1    2
[4,]    2    5    5
[5,]    1    0    1
[6,]    1    2    2
[7,]    2    4    5
[8,]    1    2    1
```

Una vez se tiene el cluster al que pertenece cada muestra, se separa la matriz siguiendo el criterio anterior.

```
> mc1=subset(m,m[,1]==1)
> mc2=subset(m,m[,1]==2)
```

Por último, limpiamos la columna introducida para el fin buscado y mostramos los dos conjuntos de datos clusterizados.

```
> (mc1=mc1[,-1])
```

```
     [,1] [,2]
[1,]    1    2
[2,]    0    1
[3,]    2    2
[4,]    2    1
```

```
> (mc2=mc2[,-1])
```

```
     [,1] [,2]
[1,]    4    4
[2,]    3    5
[3,]    5    5
[4,]    4    5
```

Se puede observar que las muestras 3,5,6,8 pertenecen al mismo grupo, mientras que las muestras 1,2,4,7 se encuentran en el restante.

2. Desarrollo por parte del alumno.