UNIVERSIDAD POLITÉCNICA DE MADRID

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE SISTEMAS INFORMÁTICOS

Máster en Software de Sistemas Distribuidos y Empotrados

Ciencia de Datos

Ejercicio 2 – Segmentación de datos – Clustering

Alejandro Casanova Martín

N.º de matrícula: bu0383

Madrid, 22 de abril 2024

Índice

1. Preparación de datos

1) Cargar en R los datos del archivo *Adult-data.csv* en el frame *Frame0*. Los datos proceden de la página *https://archive.ics.uci.edu/ml/datasets.php* (UCI Machine Learning Repository).

```
> script_dir <- "C:/Users/alex/Desktop/Máster Software Embebido/2
   Segundo Semestre/2 Ciencia de Datos/Ejercicios" # Actualizar con el
   directorio correcto
> setwd(script_dir); getwd()
[1] "C:/Users/alex/Desktop/Máster Software Embebido/2 Segundo
   Semestre/2 Ciencia de Datos/Ejercicios"
> Frame0 <- as.data.frame(read.csv("Ficheros/Adult-data.csv",
   header=TRUE, sep=',', encoding='latin1'))
> head(Frame0)
 Edad  Tipo.de.trabajo Peso.final Educacion Educacion.num.años ...
1  39        State-gov     77516  Bachelors                13 ...
2  50 Self-emp-not-inc     83311  Bachelors                13 ...
3  38          Private    215646    HS-grad                 9 ...
4  53          Private    234721       11th                 7 ...
5  28          Private    338409  Bachelors                13 ...
6  37          Private    284582    Masters                14 ...
```

2) Extraer en el subframe SubFrame0 las variables 'Edad', 'Educacion-num-años', 'Raza' y 'Nivel de ingresos'.

```
> SubFrame0 <-subset.data.frame(Frame0, select=c('Edad',
   'Educacion.num.años', 'Raza', 'Nivel.de.ingresos'))
> head(SubFrame0)
  Edad Educacion.num.años  Raza Nivel.de.ingresos
1   39                13 White             <=50K
2   50                13 White             <=50K
3   38                 9 White             <=50K
4   53                 7 Black             <=50K
5   28                13 Black             <=50K
6   37                14 White             <=50K
```

3) Determinar si en *SubFrame0* existen campos no definidos (con contenido ?).

```
> subset.data.frame(SubFrame0, Edad == "?" | Educacion.num.años == "?"
   | Raza == "?" | Nivel.de.ingresos == "?")
[1] Edad       Educacion.num.años Raza          Nivel.de.ingresos
<0 rows> (or 0-length row.names)
```

No es encontraron campos no definidos.

4) Obtener el *SubFrame1* eliminando los registros de *SubFrame0* con algún campo no definido.

```
> SubFrame1 <- subset.data.frame(SubFrame0, Edad != "?" &
   Educacion.num.años != "?" & Raza != "?" & Nivel.de.ingresos != "?")
```

En este caso no es necesario, dado que no hay campos indefinidos.

5) Obtener los rangos de variación de las variables *'Edad'* y *'Educacion-num-años'*.

```
> range(SubFrame1$Edad); range(SubFrame1$Educacion.num.años)
[1] 17 90
[1]  1 16
```

6) Obtener los valores que pueden tomar las variables *'Raza'* y *'Nivel de ingresos'*.

```
> race_names <- unique(SubFrame1$Raza)
> income_classes <- unique(SubFrame1$Nivel.de.ingresos)
> race_names
[1] "White" "Black" "Asian-Pac-Islander" "Amer-Indian-Eskimo"
[4] "Other"
> income_classes
[1] "<=50K" ">50K"
```

7) Obtener la tabla de frecuencias de la variable *'Raza'*.

```
> table(SubFrame1$Raza)
Amer-Indian-Eskimo Asian-Pac-Islander Black Other White
               311               1039  3124   271 27816
```

8) Queremos hacer un clustering de las variables *'Edad'*, *'Educacion-num años'* y *'Nivel de ingresos'*, pero no queremos perder la información de la variable *'Raza'*. Obtener el *SubFrame2* redefiniendo numéricamente la variable *'Nivel de ingresos'* para que sea representativa en el clustering y la variable *'Raza'* para que apenas influya en el clustering.

```
> race_to_num_mapping <- setNames(1:length(race_names)*1e-8,
   race_names)
> income_to_num_mapping <- setNames(1:length(income_classes)*0.5,
   income_classes)
> SubFrame2 <- SubFrame1
> SubFrame2$Raza <- sapply(SubFrame2$Raza, function(x)
   race_to_num_mapping[as.character(x)])
> range(SubFrame2$Raza)
[1] 1e-08 5e-08
> SubFrame2$Nivel.de.ingresos <- sapply(SubFrame2$Nivel.de.ingresos,
   function(x) income_to_num_mapping[as.character(x)])
> apply(SubFrame2, 2, range)
     Edad Educacion.num.años  Raza Nivel.de.ingresos
[1,]   17                  1 1e-08               0.5
[2,]   90                 16 5e-08               1.0
> head(SubFrame2)
  Edad Educacion.num.años  Raza Nivel.de.ingresos
1   39                 13 1e-08               0.5
2   50                 13 1e-08               0.5
3   38                  9 1e-08               0.5
4   53                  7 2e-08               0.5
5   28                 13 2e-08               0.5
6   37                 14 1e-08               0.5
```

Hemos mapeado la variable 'Raza' a números muy pequeños, de modo que no sea significativa. El resto de las variables serán normalizadas para quedar contenidas entre 0 y 1. A continuación convertimos los datos en una matriz numérica.

```
> kmdata_orig = as.matrix(SubFrame2[,1:4])
> kmdata <- kmdata_orig[,1:4]
> mode(kmdata) = "numeric"
> kmdata[1:10,]
   Edad Educacion.num.años  Raza Nivel.de.ingresos
1    39                 13 1e-08               0.5
2    50                 13 1e-08               0.5
3    38                  9 1e-08               0.5
4    53                  7 2e-08               0.5
5    28                 13 2e-08               0.5
6    37                 14 1e-08               0.5
7    49                  5 2e-08               0.5
```

```
8    52                  9 1e-08              1.0
9    31                 14 1e-08              1.0
10   42                 13 1e-08              1.0
```

Finalmente, normalizamos los datos de la matriz.

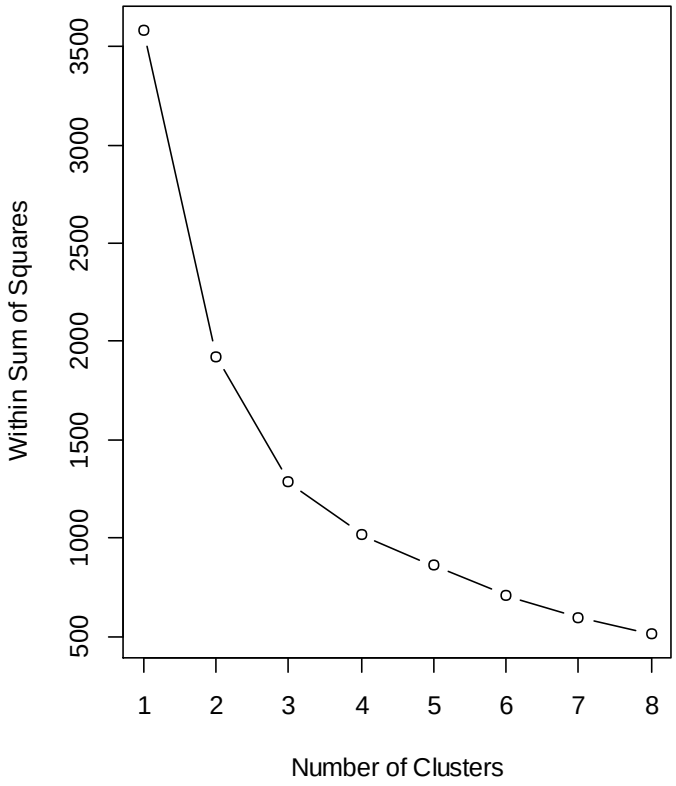
```
> max_val = numeric(2)
> min_val = numeric(2)
> for (k in 1:2) max_val[k] <- max(as.numeric(kmdata[,k]))
> for (k in 1:2) min_val[k] <- min(as.numeric(kmdata[,k]))
> kmdata[,1:2] <- scale(kmdata[,1:2], center=min_val, scale=(max_val-
   min_val))
> kmdata[1:10,]
        Edad Educacion.num.años  Raza Nivel.de.ingresos
1  0.3013699          0.8000000 1e-08               0.5
2  0.4520548          0.8000000 1e-08               0.5
3  0.2876712          0.5333333 1e-08               0.5
4  0.4931507          0.4000000 2e-08               0.5
5  0.1506849          0.8000000 2e-08               0.5
6  0.2739726          0.8666667 1e-08               0.5
7  0.4383562          0.2666667 2e-08               0.5
8  0.4794521          0.5333333 1e-08               1.0
9  0.1917808          0.8666667 1e-08               1.0
10 0.3424658          0.8000000 1e-08               1.0
> apply(kmdata, 2, range)
     Edad Educacion.num.años  Raza Nivel.de.ingresos
[1,]    0                  0 1e-08               0.5
[2,]    1                  1 5e-08               1.0
```

Comprobamos que los datos de la nueva matriz están normalizados entre 0 y 1.

2. Clustering y análisis de los resultados

a) Calcular mediante el criterio *elbow* el valor adecuado del número de clusters k.

```
> wss <- numeric(8)
> for (k in 1:8) wss[k] <- (sum(kmeans(kmdata, centers=k,
   nstart=25)$withinss))
> plot(1:8, wss, type="b", xlab="Number of Clusters", ylab="Within Sum
   of Squares")
```



9) Obtener con dicho valor de k el clustering correspondiente.

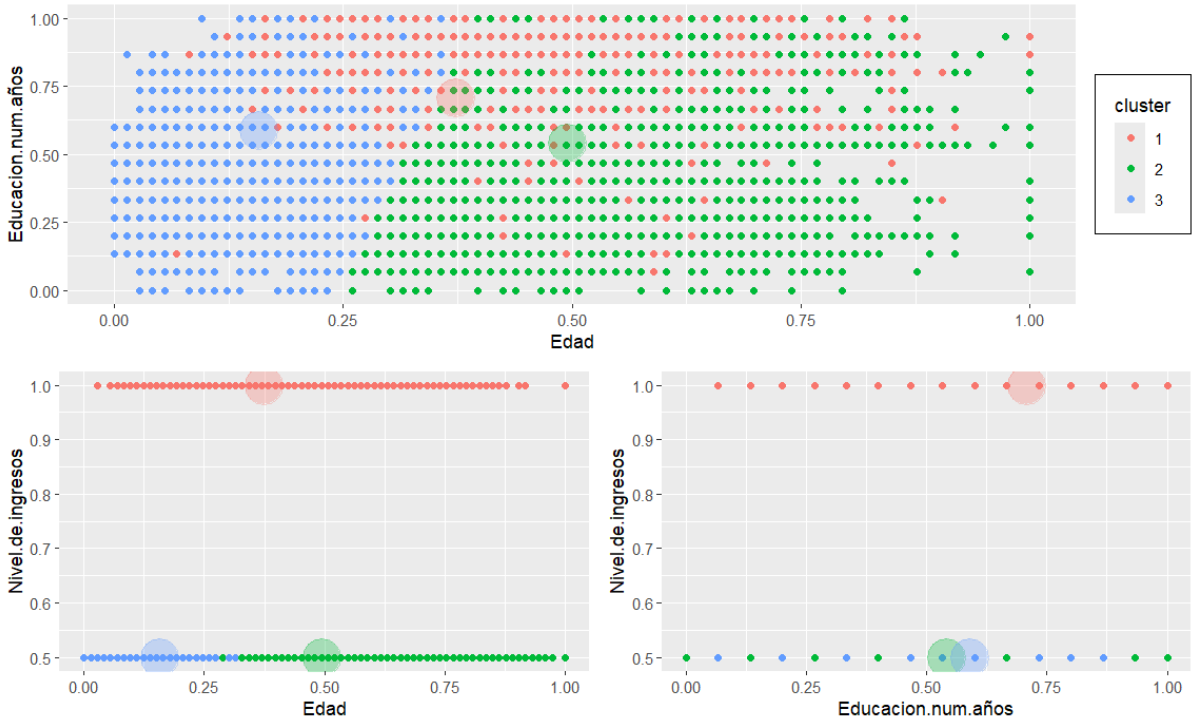
Hemos escogido k = 3

```
> km3 = kmeans(kmdata,3, nstart=25)
> km3
K-means clustering with 3 clusters of sizes 7841, 16390, 8330

Cluster means:
       Edad Educacion.num.años         Raza  Nivel.de.ingresos
1 0.3732855          0.7074438 1.146282e-08                1.0
2 0.1574365          0.5888550 1.253386e-08                0.5
3 0.4944761          0.5418167 1.230372e-08                  0.5

Clustering vector:
   1    2    3    4    5    6    7    8    9   10
   2    3    2    3    2    2    3    1    1    1
  11   12   13   14   15   16   17   18   19   20
   1    1    2    2    1    2    2    2    2    1
  21   22   23   24   25   26   27   28   29   30
   1    3    2    3    3    1    2    1    2    3
  31   32   33   34   35   36   37   38   39   40
   2    2    3    2    2    3    2    2    1    3
```

```
  41   42   43   44   45   46   47   48   49   50
...

Within cluster sum of squares by cluster:
[1] 361.0121 503.2563 420.9180
 (between_SS / total_SS =  64.1 %)

Available components:

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

10) Obtener las gráficas bidimensionales del clustering 'Edad'-'Educacion-num-años', 'Edad'-'Nivel de ingresos' y 'Educacion-num años'-'Nivel de ingresos'.

```
> NUM_CLUSTERS <- 3
> km_selected <- km3
> df <- as.data.frame(kmdata)
> df$cluster <- factor(km_selected$cluster)
> centers <- as.data.frame(km_selected$centers)
> library(ggplot2)
> library(grid) #gráficos en cuadrícula
> library(gridExtra)
> g1 <- ggplot(data=df, aes(x=Edad, y=Educacion.num.años,
   color=cluster )) +
   geom_point() +
   geom_point(data=centers, aes(x=Edad, y=Educacion.num.años,
   color=as.factor(1:NUM_CLUSTERS)), size=10, alpha=.3,
   show.legend=FALSE)
> g2 <- ggplot(data=df, aes(x=Edad, y=Nivel.de.ingresos,
   color=cluster )) +   geom_point() + theme(legend.position="none") +
   geom_point(data=centers, aes(x=Edad,y=Nivel.de.ingresos,
   color=as.factor(1:NUM_CLUSTERS)), size=10, alpha=.3,
   show.legend=FALSE)
> g3 <- ggplot(data=df, aes(x=Educacion.num.años, y=Nivel.de.ingresos,
   color=cluster )) +
   geom_point() + theme(legend.position="none") +
   geom_point(data=centers, aes(x=Educacion.num.años,
   y=Nivel.de.ingresos, color=as.factor(1:NUM_CLUSTERS)), size=10,
   alpha=.3, show.legend=FALSE)
> grid.arrange(
   arrangeGrob(g1 + theme(
     legend.box.background = element_rect(),
     legend.box.margin = margin(6, 6, 6, 6))),
   arrangeGrob(g2, g3, ncol = 2)
)
```



11) A la vista de las gráficas, describir las propiedades características de cada uno de los clusters obtenidos.

Cluster 1 (rojo): personas con nivel de ingresos alto y, generalmente nivel de estudios alto.

Cluster 2 (verde): personas con nivel de ingresos bajo y edad generalmente por encima de los 40 años.

Cluster 3 (azul): personas con nivel de ingresos bajo y edad generalmente por debajo de los 40 años.

Se puede observar que el cluster 1 apenas contiene personas con un nivel de estudios bajo, mientras que los cluster 2 y 3 contienen personas con nivel de ingresos bajo, pero nivel de estudios muy variado y con la media centrada. Esto nos da a entender que probablemente haya otras variables, que no hemos tenido en cuenta y adicionales al nivel educativo, que favorecen la pertenencia a los clusters 2 y 3.

12) Calcular para cada uno de los clusters la tabla de frecuencia de la variable 'Raza'.

```
> df_cluster1<-subset.data.frame(df, cluster == 1)$Raza
> df_cluster1 <- sapply(df_cluster1, function(x)
    names(race_to_num_mapping[as.numeric(round(x*1e08))]))
> df_cluster2<-subset.data.frame(df, cluster == 2)$Raza
> df_cluster2 <- sapply(df_cluster2, function(x)
    names(race_to_num_mapping[as.numeric(round(x*1e08))]))
> df_cluster3<-subset.data.frame(df, cluster == 3)$Raza
> df_cluster3 <- sapply(df_cluster3, function(x)
    names(race_to_num_mapping[as.numeric(round(x*1e08))]))
> table(df_cluster1); round(prop.table(table(df_cluster1))*100, 2) #
    Cluster 1 (<=50k)
df_cluster1
Amer-Indian-Eskimo Asian-Pac-Islander   Black   Other     White
                36                276     387      25      7117
df_cluster1
```

```
Amer-Indian-Eskimo Asian-Pac-Islander   Black   Other     White
              0.46               3.52    4.94    0.32     90.77
> table(df_cluster2); round(prop.table(table(df_cluster2))*100, 2) #
   Cluster 2 (<50k, old)
df_cluster2
Amer-Indian-Eskimo Asian-Pac-Islander   Black   Other     White
                93                227     946      60      7004
df_cluster2
Amer-Indian-Eskimo Asian-Pac-Islander   Black   Other     White
              1.12               2.73   11.36    0.72     84.08
> table(df_cluster3); round(prop.table(table(df_cluster3))*100, 2) #
   Cluster 3 (<50k, young)
df_cluster3
Amer-Indian-Eskimo Asian-Pac-Islander   Black   Other     White
               182                536    1791     186     13695
df_cluster3
Amer-Indian-Eskimo Asian-Pac-Islander   Black   Other     White
              1.11               3.27   10.93    1.13     83.56
```

13) Comparar las tablas de frecuencia anteriores con la tabla de frecuencias de la variable 'Raza' obtenida en la sección anterior.

Proporciones globales de cada raza:

```
> table(SubFrame1$Raza); round(prop.table(table(SubFrame1$Raza))*100,
   2) # Overall
Amer-Indian-Eskimo Asian-Pac-Islander   Black   Other     White
               311               1039    3124     271     27816
Amer-Indian-Eskimo Asian-Pac-Islander   Black   Other     White
              0.96               3.19    9.59    0.83     85.43
```

Proporción de pertenencia al cluster 1 por raza:

```
> round(table(df_cluster1) / table(SubFrame1$Raza) * 100, 2) # High
   income percentage per race
df_cluster1
Amer-Indian-Eskimo Asian-Pac-Islander   Black   Other     White
             11.58              26.56   12.39   9.23      25.59
```

Se puede observar que las razas 'White' y 'Asian-Pac-Islander' son las que presentan un mayor porcentaje de ingresos altos. Respectivamente el 25,59% y el 26,56% del total de personas pertenecientes a estas razas reciben ingresos altos (>50k). Por otro lado, para el resto de las razas, sólo en torno al 10% de las personas alcanzan dicho nivel.

A continuación se calculan los incrementos en las proporciones de cada raza para cada cluster, con respecto a las proporciones globales.

```
> round((prop.table(table(df_cluster1)) -
   prop.table(table(SubFrame1$Raza))) /
   prop.table(table(SubFrame1$Raza))*100, 2)
df_cluster1
Amer-Indian-Eskimo Asian-Pac-Islander    Black  Other     White
            -51.93              10.31   -48.56 -61.69      6.25
```

En el cluster 1 (ingresos altos y nivel de estudios alto), hay un aumento de la proporción de las personas de razas "White" y 'Asian-Pac-Islander', un 6,25% y 10,31% respectivamente, mientras que la proporción del resto de razas disminuye drásticamente (hasta un 61,69% en el caso de las razas 'Other' y en torno a un 50% en las razas 'Black' y 'Amer-Indian-Eskimo'.

```
> round((prop.table(table(df_cluster2)) -
   prop.table(table(SubFrame1$Raza))) /
   prop.table(table(SubFrame1$Raza))*100, 2)
df_cluster2
Amer-Indian-Eskimo Asian-Pac-Islander    Black  Other     White
             16.89             -14.60    18.37 -13.46     -1.58
```

En el cluster 2 (ingresos bajos y edad por encima de los 40 años), aumentan en torno a un 18% las proporciones de las razas 'Black' y 'Amer-Indian-Eskimo', y disminuyen las de las razas 'Other' y 'Asian-Pac-Islander' en torno a un 14%. La proporción de la raza 'White' disminuye ligeramente (un 1.58%), aunque hay que considerar que es la raza más frecuente globalmente.

```
> round((prop.table(table(df_cluster3)) -
   prop.table(table(SubFrame1$Raza))) /
   prop.table(table(SubFrame1$Raza))*100, 2)
df_cluster3
Amer-Indian-Eskimo Asian-Pac-Islander    Black  Other     White
             16.26               2.49    13.89  36.35     -2.19
```

Finalmente, en el cluster 3 (ingresos bajos y edad por debajo de los 40 años) aumentan las proporciones de todas las razas salvo la blanca, que disminuye ligeramente (un 2.19%). Igualmente tendremos en cuenta que la frecuencia de esta raza es la mayor globalmente.

Como conclusión, hemos observado que las razas más favorecidas económicamente, y con mayor nivel de estudios son 'Asian-Pac-Islander' y 'White', frente a las razas 'Amer-Indian-Eskimo', 'Black' y 'Other' que presentan generalmente un nivel de ingresos y de estudios menor. La edad no parece tener una influencia tan relevante en el nivel de ingresos como sí el nivel educativo y la raza.