Análisis de clasificación de la renta per cápita disponible de los municipios de Madrid

Sara Bengoechea Rodríguez e Inés Martínez Pereda

11/17/2020

Introducción

El presente proyecto tiene como objetivo llevar a cabo un análisis de clasificación de la renta per cápita disponible de los municipios de Madrid. En este se desarrollarán distintos modelos para predecir si dichos municipios tendrán una renta per cápita superior o inferior a la media.

Importación de las librerías necesarias

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(readxl)
library(skimr)
library(MASS) # Para LDA
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(klaR) # Para gráficos de partición
```

Importación y visualización de datos

Importación de datos y visualización de columnas.

Las variables son:

-Indicador de renta disponible bruta municipal per cápita (euros) -Paro registrado por 100 hab -Valor catastral por unidad urbana

-Población empadronada
-Comercio y hostelería (Ocupados por 1.000 h)
-Administración pública, educación y sanidad
-Centros escolares (2012)
-Densidad de edificios por km^2 -Energía eléctrica facturada per cápita (KW/hora)

```
data <- read_excel("data.xlsx")
colnames(data)
```

```
##  [1] "Municipios"
##  [2] "Indicador de renta disponible bruta municipal per cápita (euros) (a)"
##  [3] "Paro registrado por 100 hab (1)"
##  [4] "Valor catastral por unidad urbana (2)"
##  [5] "Población empadronada"
##  [6] "Comercio y hostelería (Ocupados por 1.000 h (a))"
##  [7] "Administración pública, educación y sanidad"
##  [8] "Centros escolares (2102)"
##  [9] "Densidad de edificios por km^2"
## [10] "Energía eléctrica facturada per cápita (KW/hora)"
```

Cambiamos el nombre de las variables.

```
names(data)[2] <- "renta"
names(data)[3] <- "paro_por_100hab"
names(data)[4] <- "valor_catastral"
names(data)[5] <- "poblacion"
names(data)[6] <- "comercio_hosteleria"
names(data)[7] <- "administracion_publica_educación_sanidad"
names(data)[8] <- "colegios"
names(data)[9] <- "edificios_densidad"
names(data)[10] <- "energia_per_capita"
colnames(data)
```

```
##  [1] "Municipios"
##  [2] "renta"
##  [3] "paro_por_100hab"
##  [4] "valor_catastral"
##  [5] "poblacion"
##  [6] "comercio_hosteleria"
##  [7] "administracion_publica_educación_sanidad"
##  [8] "colegios"
##  [9] "edificios_densidad"
## [10] "energia_per_capita"
```

Análisis exploratorio de la renta per cápita por municipios y definición de la variable target

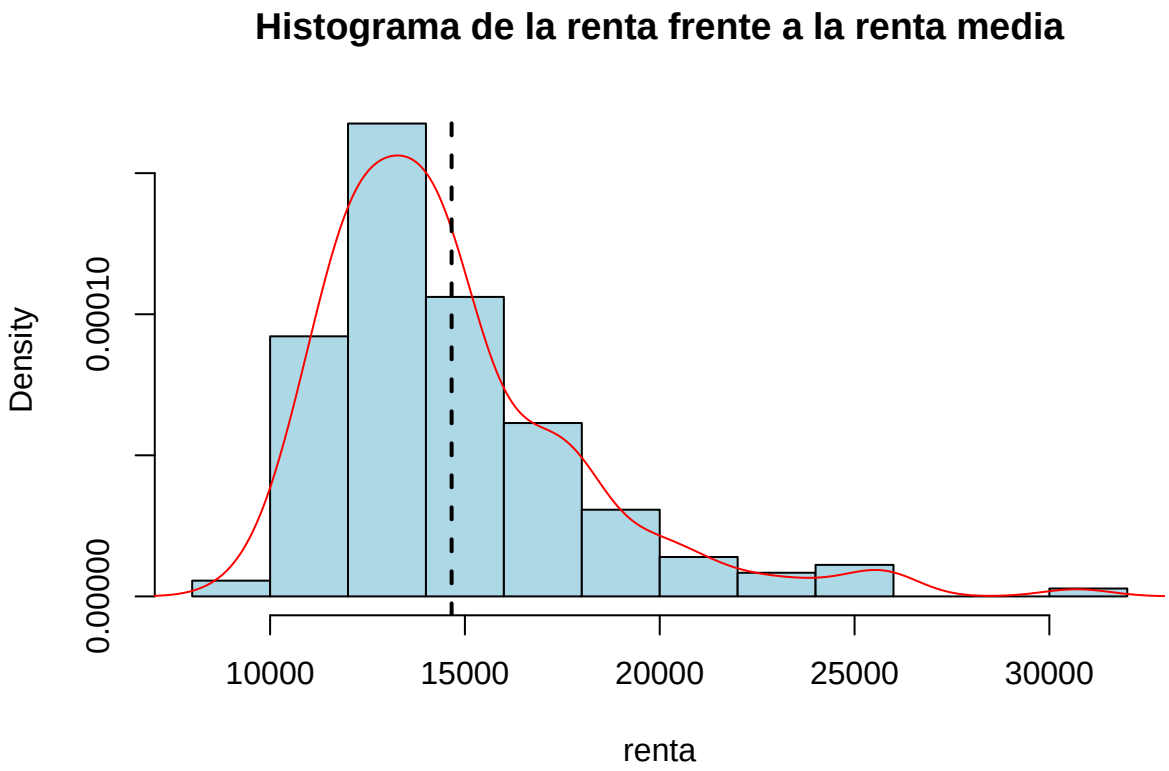
Los estadísticos más relevantes de nuestra variable target son los siguientes.

```
summary(data$renta)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    9120   12408   13890   14658   16114   30701
```

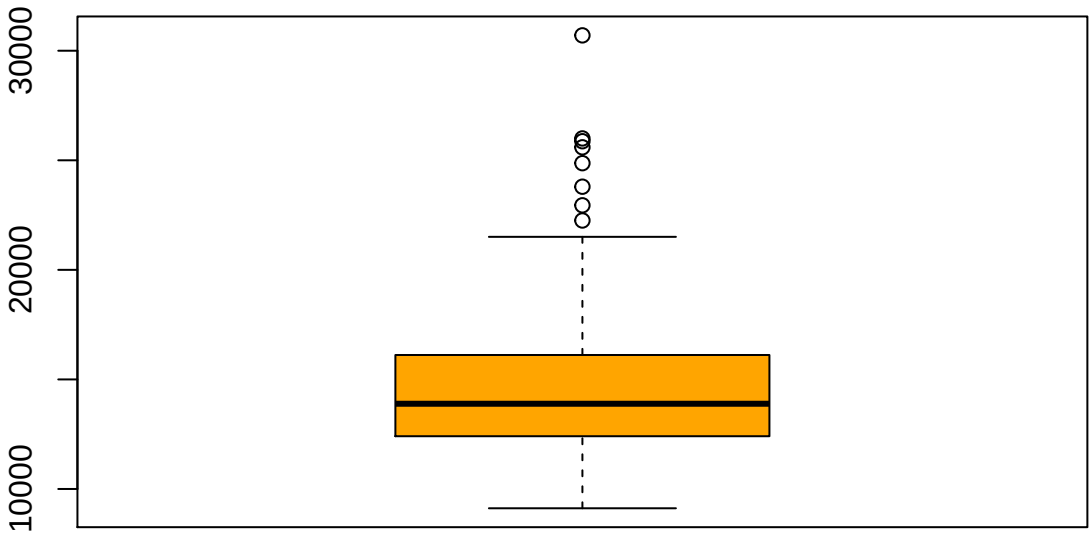
En el histograma se puede ver cómo la mayor parte de valores está por debajo de la renta media. No sigue una distribución normal, es asimétrica hacia la derecha.

```r
attach(data)
hist(renta, probability = T, col = "light blue",main = "Histograma de la renta frente a la renta media")
lines(density(renta), col ="red")
abline(v = mean(renta), lwd = 2, lty = "dashed")
```



Con el boxplot vemos la gran dispersión de esta variable y los muchos outliers superiores que hay, que distorsionan la renta media.

```r
boxplot(renta, col = "orange")
```



Con la media de la renta establecemos la variable dummy que toma los valores “superior” e “inferior” y visualizamos el resultado.

```r
u <- mean(data$renta) # Hayamos la renta media per cápita de los municipios

data$renta_media <- ifelse(data[2] > u, "superior", "inferior") # Creamos dummy

data <- data %>% mutate(renta_media = factor(renta_media, levels = c("superior", "inferior"))) # conver

head(data)
```

```
## # A tibble: 6 x 11
##   Municipios  renta paro_por_100hab valor_catastral poblacion comercio_hostel~
##   <chr>       <dbl>           <dbl>           <dbl>     <dbl>            <dbl>
## 1 Acebeda (~ 12657.           11.9             23.4        67             78.1
## 2 Ajalvir    16854.            7.53            82.8      4261            206.
## 3 Alameda d~ 15456.            6.05            45.3       248            131.
## 4 Alamo (El) 13145.           11.3             39.6      8845             47.2
## 5 Alcalá de~ 14848.           10.7            125.     204823             85.9
## 6 Alcobendas 22253.            7.48           203.     112196            223.
## # ... with 5 more variables: administracion_publica_educación_sanidad <dbl>,
## #   colegios <dbl>, edificios_densidad <dbl>, energia_per_capita <dbl>,
## #   renta_media <fct>
```

Los datos están divididos en un 36% de municipios cuya renta per cápita es superior a la media frente a un 64% que es inferior.

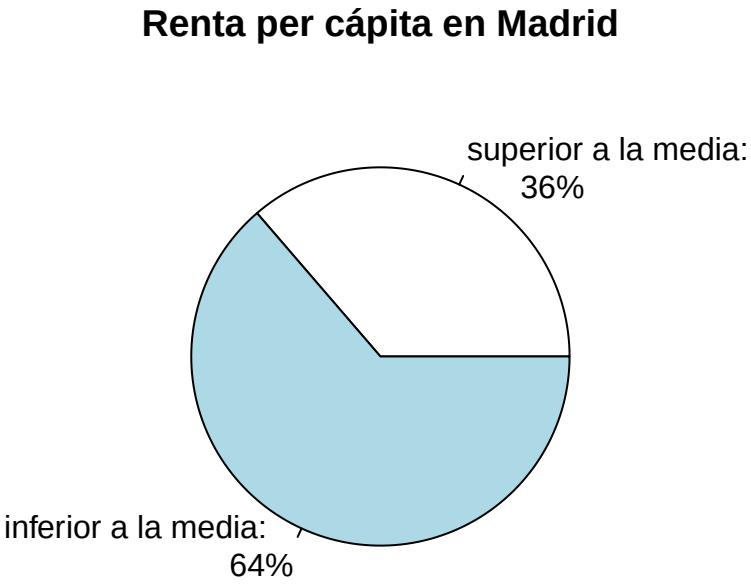
```r
count_renta <- data %>% count(renta_media) # Creamos dataframe para obtener % de renta superior e infer

count_renta["extra"] <- "a" # añadimos col extra


count_renta <- group_by(count_renta, extra) %>% mutate(percent = round(n/sum(n),2))
count_renta["extra"] <- NULL
count_renta
```

```
## # A tibble: 2 x 3
##   renta_media     n percent
##   <fct>       <int>   <dbl>
## 1 superior       65    0.36
## 2 inferior      114    0.64
```

```r
library(plotrix)
slices <- c(65, 114)
lbls <- c("superior a la media:
      36%", "inferior a la media:
      64%")
pie(slices,labels=lbls,
   main="Renta per cápita en Madrid")
```



Eliminamos columna de la renta numérica

```
data["renta"] <- NULL  # Eliminamos la columna numérica de renta
head(data)
```

```
## # A tibble: 6 x 10
##   Municipios paro_por_100hab valor_catastral poblacion comercio_hostel~
##   <chr>                <dbl>           <dbl>     <dbl>            <dbl>
## 1 Acebeda (~           11.9             23.4        67             78.1
## 2 Ajalvir               7.53            82.8      4261            206.
## 3 Alameda d~            6.05            45.3       248            131.
## 4 Alamo (El)           11.3             39.6      8845             47.2
## 5 Alcalá de~           10.7            125.     204823             85.9
## 6 Alcobendas            7.48           203.     112196            223.
## # ... with 5 more variables: administracion_publica_educación_sanidad <dbl>,
## #   colegios <dbl>, edificios_densidad <dbl>, energia_per_capita <dbl>,
## #   renta_media <fct>
```

Selección de variables meidante step AIC

Mediante el procedimiento de step_aic (direction = "both"), obtenemos que el mejor modelo que contiene un total de 6 variables. De las 8 variables iniciales con las que contábamos, todas ayudan a predecir en nuestro modelo excepto comercio_hosteleria y edificios_densidad.

```
# Creamos un modelo con todas las varibles excepto Municipios

model_all <- glm(renta_media ~. -Municipios, family = binomial(link = logit), data = data)
summary(model_all)
```

```
##
## Call:
## glm(formula = renta_media ~ . - Municipios, family = binomial(link = logit),
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
```

```
## -2.2983  -0.3853   0.1366   0.4944   2.0239
##
## Coefficients:
##                                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)                             -6.305e+00  1.678e+00  -3.756 0.000172
## paro_por_100hab                          7.888e-01  1.426e-01   5.530 3.21e-08
## valor_catastral                         -2.118e-02  6.075e-03  -3.487 0.000488
## poblacion                                6.928e-05  3.931e-05   1.762 0.078003
## comercio_hosteleria                      7.697e-03  7.520e-03   1.024 0.306020
## administracion_publica_educación_sanidad 1.486e-02  6.807e-03   2.183 0.029045
## colegios                                -1.588e-01  8.788e-02  -1.807 0.070815
## edificios_densidad                      -2.816e-03  5.369e-03  -0.524 0.600017
## energia_per_capita                       1.661e-04  1.824e-04   0.911 0.362536
##
## (Intercept)                             ***
## paro_por_100hab                         ***
## valor_catastral                         ***
## poblacion                               .
## comercio_hosteleria
## administracion_publica_educación_sanidad *
## colegios                                .
## edificios_densidad
## energia_per_capita
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 234.56  on 178  degrees of freedom
## Residual deviance: 116.30  on 170  degrees of freedom
## AIC: 134.3
##
## Number of Fisher Scoring iterations: 6
```

```
stepAIC(model_all, direction = "both")
```

```
## Start:  AIC=134.3
## renta_media ~ (Municipios + paro_por_100hab + valor_catastral +
##     poblacion + comercio_hosteleria + administracion_publica_educación_sanidad +
##     colegios + edificios_densidad + energia_per_capita) - Municipios
##
##                                         Df Deviance    AIC
## - edificios_densidad                     1   116.58 132.58
## - energia_per_capita                     1   117.22 133.22
## - comercio_hosteleria                    1   117.36 133.37
## <none>                                       116.30 134.30
## - poblacion                              1   119.14 135.13
## - colegios                               1   119.56 135.56
## - administracion_publica_educación_sanidad  1   123.57 139.57
## - valor_catastral                        1   130.29 146.29
## - paro_por_100hab                        1   169.06 185.06
##
## Step:  AIC=132.58
## renta_media ~ paro_por_100hab + valor_catastral + poblacion +
##     comercio_hosteleria + administracion_publica_educación_sanidad +
```

```
##     colegios + energia_per_capita
##
##                                           Df Deviance    AIC
## - comercio_hosteleria                      1   117.45 131.45
## - energia_per_capita                       1   117.86 131.86
## <none>                                         116.58 132.58
## - poblacion                                1   119.90 133.90
## + edificios_densidad                       1   116.30 134.30
## - colegios                                 1   120.65 134.65
## - administracion_publica_educación_sanidad 1   124.32 138.32
## - valor_catastral                          1   134.90 148.90
## - paro_por_100hab                          1   169.08 183.08
##
## Step:  AIC=131.45
## renta_media ~ paro_por_100hab + valor_catastral + poblacion +
##     administracion_publica_educación_sanidad + colegios + energia_per_capita
##
##                                           Df Deviance    AIC
## <none>                                         117.45 131.45
## - poblacion                                1   120.48 132.48
## + comercio_hosteleria                      1   116.58 132.58
## - colegios                                 1   121.12 133.12
## + edificios_densidad                       1   117.36 133.37
## - energia_per_capita                       1   121.44 133.44
## - administracion_publica_educación_sanidad 1   124.88 136.88
## - valor_catastral                          1   135.02 147.02
## - paro_por_100hab                          1   169.08 181.08
##
## Call:  glm(formula = renta_media ~ paro_por_100hab + valor_catastral +
##     poblacion + administracion_publica_educación_sanidad + colegios +
##     energia_per_capita, family = binomial(link = logit), data = data)
##
## Coefficients:
##                               (Intercept)
##                                 -6.293e+00
##                             paro_por_100hab
##                                   7.677e-01
##                             valor_catastral
##                                  -2.072e-02
##                                  poblacion
##                                   7.051e-05
## administracion_publica_educación_sanidad
##                                   1.523e-02
##                                   colegios
##                                  -1.618e-01
##                           energia_per_capita
##                                   2.763e-04
##
## Degrees of Freedom: 178 Total (i.e. Null);  172 Residual
## Null Deviance:       234.6
## Residual Deviance: 117.4     AIC: 131.4
```

Por lo que eliminamos de nuestro dataset las dos variables que no necesitamos

```r
data$comercio_hosteleria <- NULL
data$edificios_densidad <- NULL
colnames(data)
```

```
## [1] "Municipios"
## [2] "paro_por_100hab"
## [3] "valor_catastral"
## [4] "poblacion"
## [5] "administracion_publica_educación_sanidad"
## [6] "colegios"
## [7] "energia_per_capita"
## [8] "renta_media"
```

Análisis exploratorio de las variables seleccionadas frente a la variable target

Tras observar los boxplots, Las variables donde, a nivel visual, es significativo el nivel de renta per cápita, son el paro por cada 100 habitantes, el valor catastral y la administración pública, educación y sanidad.

```r
summary(data)
```

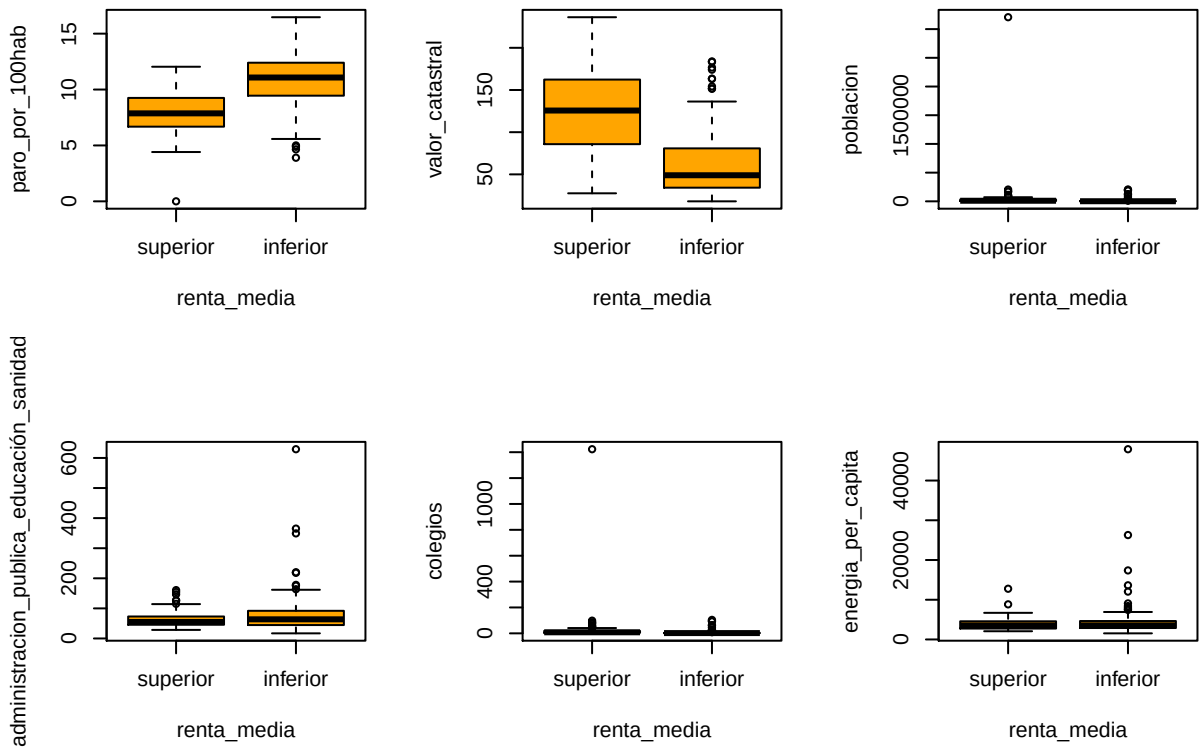
```
##   Municipios        paro_por_100hab  valor_catastral    poblacion
##  Length:179         Min.   : 0.000   Min.   : 18.06   Min.   :     51
##  Class :character   1st Qu.: 7.820   1st Qu.: 40.20   1st Qu.:    887
##  Mode  :character   Median : 9.740   Median : 76.32   Median :   3495
##                     Mean   : 9.727   Mean   : 86.11   Mean   :  36288
##                     3rd Qu.:11.620   3rd Qu.:125.40   3rd Qu.:  12207
##                     Max.   :16.470   Max.   :236.26   Max.   :3207247
##  administracion_publica_educación_sanidad    colegios      energia_per_capita
##  Min.   : 17.03                           Min.   :   0.0   Min.   : 1541
##  1st Qu.: 44.62                           1st Qu.:   1.0   1st Qu.: 2868
##  Median : 60.59                           Median :   3.0   Median : 3534
##  Mean   : 76.47                           Mean   :  18.8   Mean   : 4450
##  3rd Qu.: 88.33                           3rd Qu.:   9.0   3rd Qu.: 4617
##  Max.   :629.03                           Max.   :1423.0   Max.   :47884
##    renta_media
##  superior: 65
##  inferior:114
##
##
##
##
```

```r
attach(data)
```

```
## The following objects are masked from data (pos = 4):
##
##     administracion_publica_educación_sanidad, colegios,
##     energia_per_capita, Municipios, paro_por_100hab, poblacion,
##     valor_catastral
```

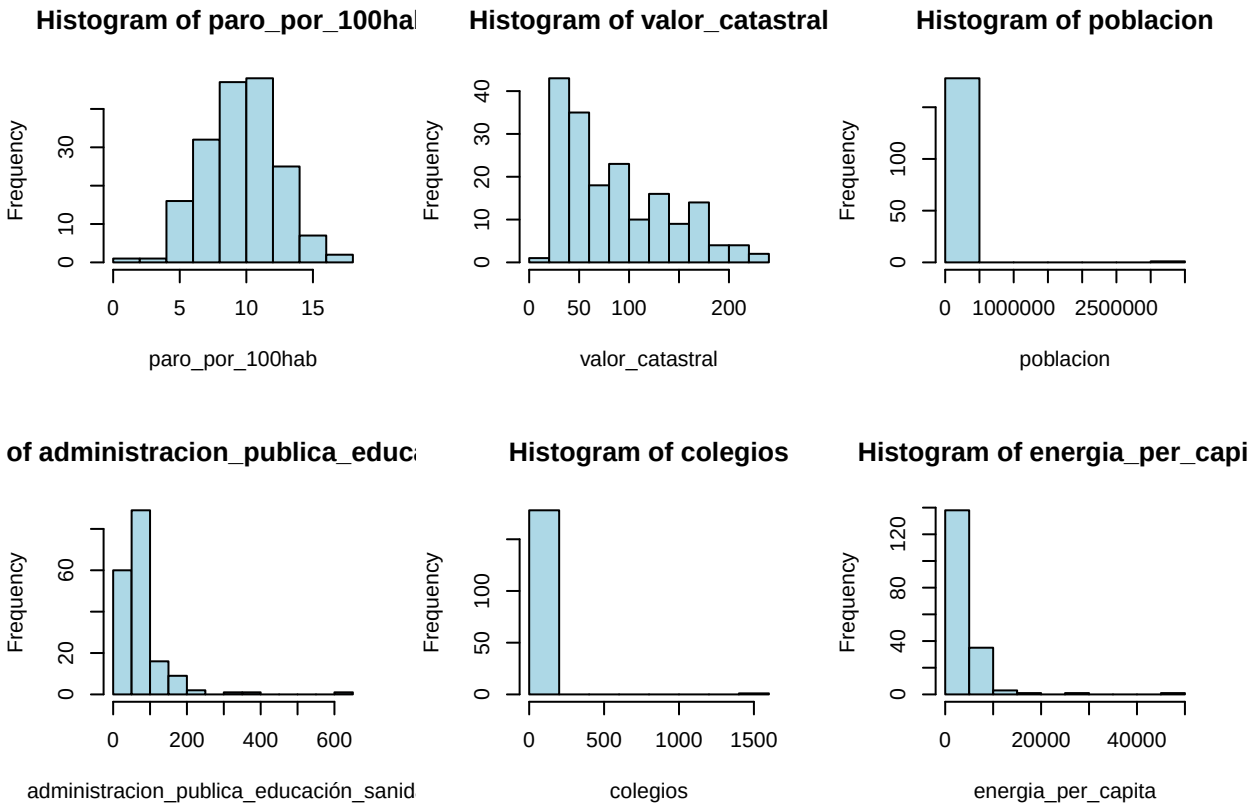
```r
par(mfrow = c(2,3))
boxplot(paro_por_100hab~renta_media, col = "orange")
boxplot(valor_catastral~renta_media, col = "orange")
boxplot(poblacion~renta_media, col = "orange")
```

```r
boxplot(administracion_publica_educación_sanidad~renta_media, col = "orange")
boxplot(colegios~renta_media, col = "orange")
boxplot(energia_per_capita~renta_media, col = "orange")
```



Podemos ver que la única variable que sigue una distribución normal es el paro, mientras que las demás son asimétricas hacia la derecha. Además, se pueden apreciar outliers en las variables de la población, administración pública, educación y sanidad, colegios y la energía per cápita.

```r
par(mfrow = c(2,3))
hist(paro_por_100hab, col = "light blue")
hist(valor_catastral, col = "light blue")
hist(poblacion, col = "light blue")
hist(administracion_publica_educación_sanidad, col = "light blue")
hist(colegios, col = "light blue")
hist(energia_per_capita, col = "light blue")
```



Regresion logistica(logit)

```
model_RL <- glm(formula = renta_media ~ paro_por_100hab + valor_catastral +
    poblacion + administracion_publica_educación_sanidad + colegios +
    energia_per_capita, family = binomial(link = logit), data = data)

summary(model_RL)
```

```
##
## Call:
## glm(formula = renta_media ~ paro_por_100hab + valor_catastral +
##     poblacion + administracion_publica_educación_sanidad + colegios +
##     energia_per_capita, family = binomial(link = logit), data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2214  -0.3875   0.1283   0.4941   1.9689
##
## Coefficients:
##                                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)                            -6.293e+00  1.673e+00  -3.761 0.000169
## paro_por_100hab                         7.677e-01  1.397e-01   5.495  3.9e-08
## valor_catastral                        -2.072e-02  5.329e-03  -3.887 0.000101
## poblacion                               7.051e-05  3.601e-05   1.958 0.050223
## administracion_publica_educación_sanidad  1.523e-02  6.831e-03   2.230 0.025767
## colegios                               -1.618e-01  7.969e-02  -2.030 0.042316
## energia_per_capita                      2.763e-04  1.528e-04   1.808 0.070611
```

```
##
## (Intercept)                               ***
## paro_por_100hab                           ***
## valor_catastral                           ***
## poblacion                                 .
## administracion_publica_educación_sanidad *
## colegios                                  *
## energia_per_capita                        .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 234.56  on 178  degrees of freedom
## Residual deviance: 117.45  on 172  degrees of freedom
## AIC: 131.45
##
## Number of Fisher Scoring iterations: 6
```

Matriz de confusion para RL

```
fit.pred <- ifelse(model_RL$fitted.values > 0.5, 1, 0)

matriz_RL <- table(fit.pred, data$renta_media)
matriz_RL
```

```
##
## fit.pred superior inferior
##        0       49       14
##        1       16      100
```

La precisión del modelo de regresión logística es del 83.24%.

```
(matriz_RL[1,1] + matriz_RL[2,2])/sum(matriz_RL)
```

```
## [1] 0.8324022
```

Ya que muy pocas de las variables son significativas, probamos a realizar un logit solo con las variables significativas. Su accuracy es ligeramente menor: 82.68% frente a 83.24%. Por lo tanto, nos quedamos con el modelo seleccionado median AIC.

```
model_RL_signif <- glm(renta_media ~ paro_por_100hab + valor_catastral + administracion_publica_educaci

summary(model_RL_signif)
```

```
##
## Call:
## glm(formula = renta_media ~ paro_por_100hab + valor_catastral +
##     administracion_publica_educación_sanidad + colegios, family = binomial(link = logit))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0678  -0.4604   0.1584   0.5108   1.9178
##
## Coefficients:
##                                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)                              -4.89848    1.41176  -3.470 0.000521
## paro_por_100hab                           0.72361    0.13165   5.497 3.87e-08
```

```
## valor_catastral                           -0.02235    0.00513  -4.356 1.32e-05
## administracion_publica_educación_sanidad  0.01479    0.00681   2.172 0.029869
## colegios                                  -0.01500    0.01033  -1.453 0.146339
##
## (Intercept)                               ***
## paro_por_100hab                           ***
## valor_catastral                           ***
## administracion_publica_educación_sanidad *
## colegios
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 234.56  on 178  degrees of freedom
## Residual deviance: 124.91  on 174  degrees of freedom
## AIC: 134.91
##
## Number of Fisher Scoring iterations: 6
```

```
fit.pred_signif <- ifelse(model_RL_signif$fitted.values > 0.5, 1, 0)

matriz_RL_signif <- table(fit.pred_signif, data$renta_media)
matriz_RL_signif
```

```
##
## fit.pred_signif superior inferior
##               0       49       15
##               1       16       99
```

```
(matriz_RL_signif[1,1] + matriz_RL_signif[2,2])/sum(matriz_RL_signif)
```

```
## [1] 0.8268156
```

Análisis de Discriminante Lineal (LDA)

```
model_LDA <- lda(renta_media ~ paro_por_100hab + valor_catastral +
    poblacion + administracion_publica_educación_sanidad + colegios +
    energia_per_capita, data = data)

model_LDA
```

```
## Call:
## lda(renta_media ~ paro_por_100hab + valor_catastral + poblacion +
##     administracion_publica_educación_sanidad + colegios + energia_per_capita,
##     data = data)
##
## Prior probabilities of groups:
##  superior  inferior
## 0.3631285 0.6368715
##
## Group means:
##          paro_por_100hab valor_catastral poblacion
## superior        7.823077       124.51092  77918.15
## inferior       10.812018        64.21684  12551.50
```

```
##          administracion_publica_educación_sanidad  colegios energia_per_capita
## superior                                 65.38908 39.292308           3931.156
## inferior                                 82.78526  7.114035           4745.076
##
## Coefficients of linear discriminants:
##                                                   LD1
## paro_por_100hab                          3.064029e-01
## valor_catastral                         -1.196099e-02
## poblacion                                2.524332e-05
## administracion_publica_educación_sanidad 4.267231e-03
## colegios                                -5.813091e-02
## energia_per_capita                       1.478378e-05
```

La matriz de confusión es la siguiente:

```
# Prediccion respuesta
ldaResult <- predict(model_LDA, newdata = data)

# Matriz de confusion
matriz_LDA <- table(ldaResult$class, data$renta_media)
matriz_LDA
```

```
##
##            superior inferior
##   superior       47       11
##   inferior       18      103
```

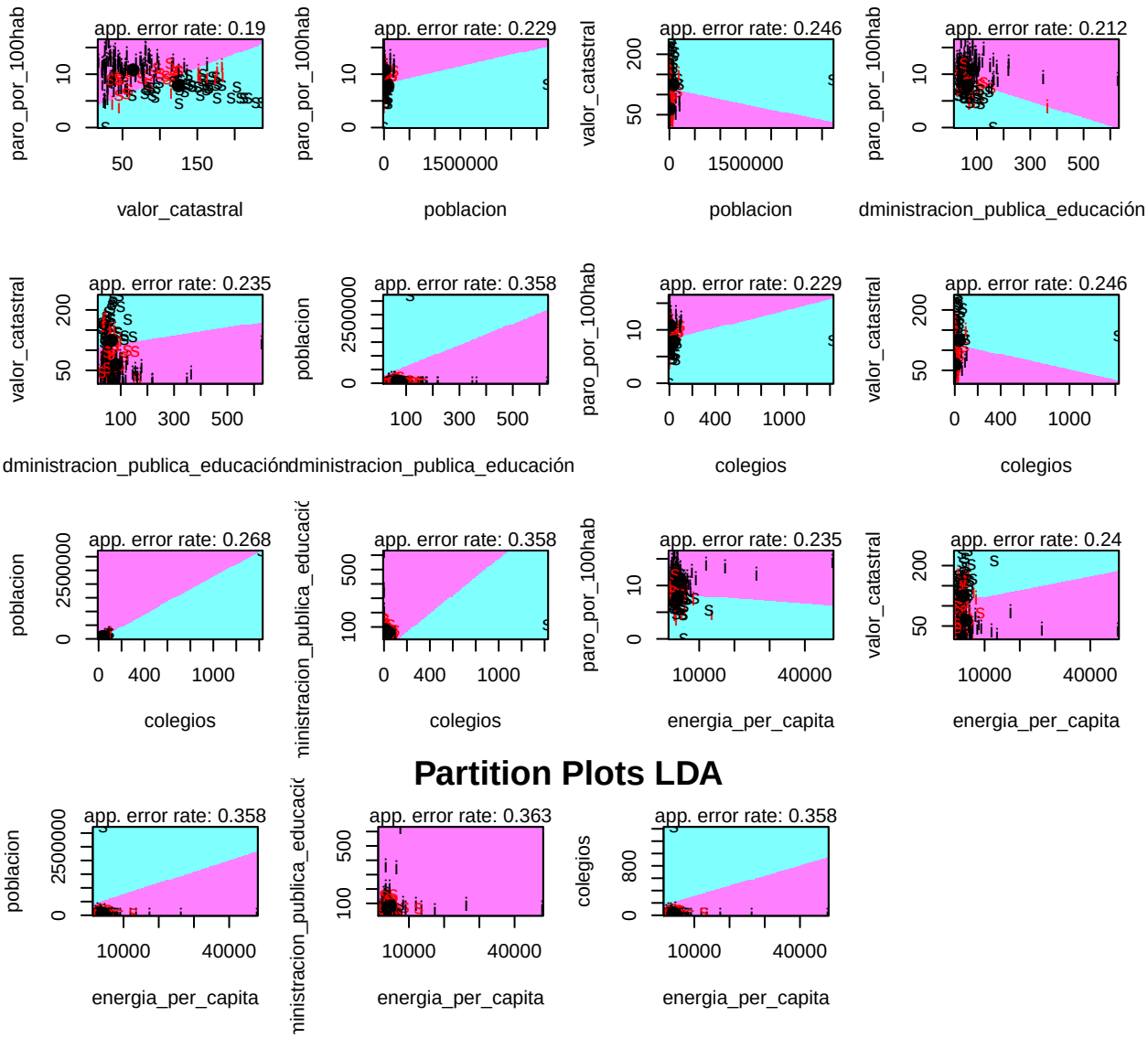
La precisión del modelo de LDA es de 83.79%, ligeramente mejor este modelo que el de regresión, un 1,11% mejor.

```
#Cálculo de la precisión del modelo de LDA
sum(diag(matriz_LDA))/sum(matriz_LDA)
```

```
## [1] 0.8379888
```

A continuación se muestran los gráficos de partición de LDA. En rojo aparecen aquellas observaciones que estarían clasificadas de manera errónea. La variable paro_por_100hab es la que tiene menor ratio de error.

```
# Graficos de particion LDA
partimat(data[,-c(1, 8)], renta_media, data=data,method="lda", main="Partition Plots LDA")
```



Análisis Discriminante Cuadrático (QDA)

```
model_QDA <- qda(renta_media ~ paro_por_100hab + valor_catastral +
    poblacion + administracion_publica_educación_sanidad + colegios +
    energia_per_capita, data = data)

model_QDA
```

```
## Call:
## qda(renta_media ~ paro_por_100hab + valor_catastral + poblacion +
##     administracion_publica_educación_sanidad + colegios + energia_per_capita,
##     data = data)
##
## Prior probabilities of groups:
##  superior  inferior
## 0.3631285 0.6368715
##
```

```
## Group means:
##          paro_por_100hab valor_catastral poblacion
## superior        7.823077       124.51092  77918.15
## inferior       10.812018        64.21684  12551.50
##          administracion_publica_educación_sanidad  colegios energia_per_capita
## superior                                65.38908 39.292308           3931.156
## inferior                                82.78526  7.114035           4745.076
```

A continuación se muestra la matriz de confusión para el modelo QDA y su precisión, que es de 79,88%. En este caso, este modelo es menos preciso que los anteriores.

```
# Prediccion respuesta
qdaResult <- predict(model_QDA, newdata = data)

# Matriz de confusion
Matriz_QDA <- table(qdaResult$class, data$renta_media)
Matriz_QDA
```

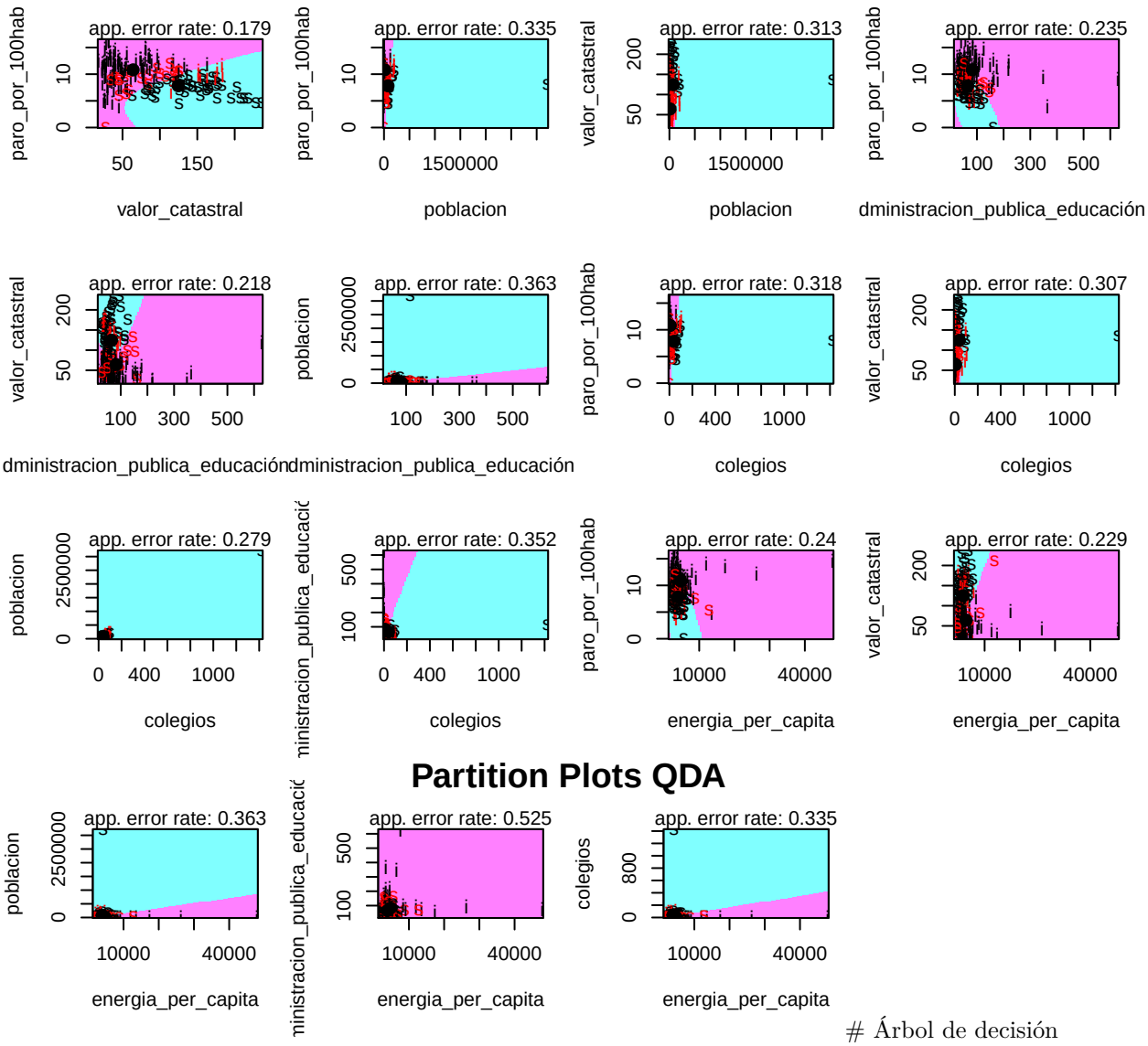
```
##
##            superior inferior
##   superior       38        9
##   inferior       27      105
```

```
sum(diag(Matriz_QDA))/sum(Matriz_QDA)
```

```
## [1] 0.7988827
```

A continuación se muestran los gráficos de partición de QDA. Como anteriormente, en rojo aparecen aquellas observaciones que estarían clasificadas de manera errónea. La variable paro_por_100hab es también la que tiene menor ratio de error.

```
partimat(data[,-c(1, 8)], renta_media, data = data, method = "qda",main = "Partition Plots QDA")
```



Árbol de decisión

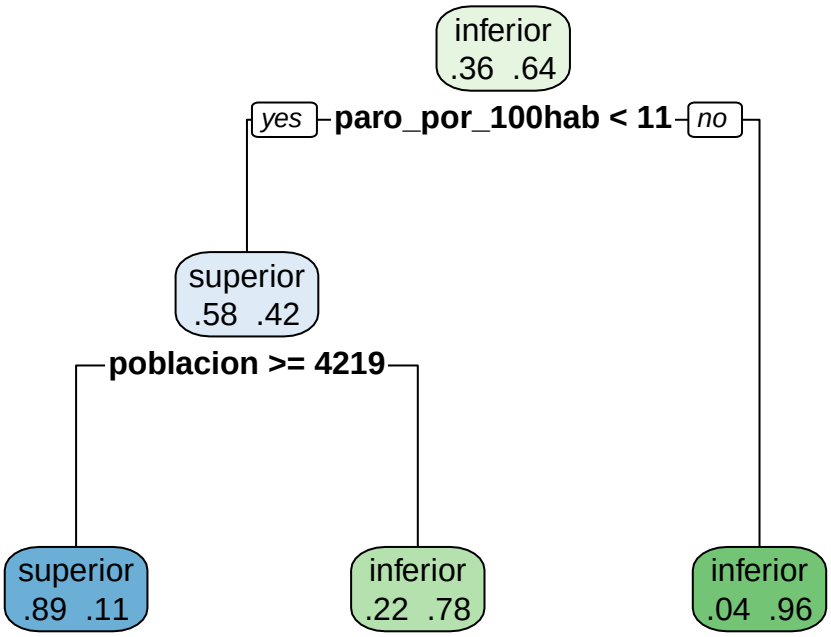
Llevamos a cabo un árbol de decisión mediante la función rpart. La interpretación de dicho árbol es la siguiente:

- Si el paro por cada 100 habitantes del municipio es superior a 11, la renta per cápita de este será inferior a la media.
- Si el paro por cada 100 habitantes es inferior a 11 y la población es menor a 4,219 este municipio tendrá una renta per cápita inferior a la media.
- Si el paro por cada 100 habitantes es inferior a 11 y la población es mayor o igual a 4219 este municipio tendrá una renta per cápita superior a la media.

```
library(rpart)
library(rpart.plot)

arbol_1 <- rpart(renta_media~.-Municipios, method = "class", data = data)

rpart.plot(arbol_1, extra = 4)
```

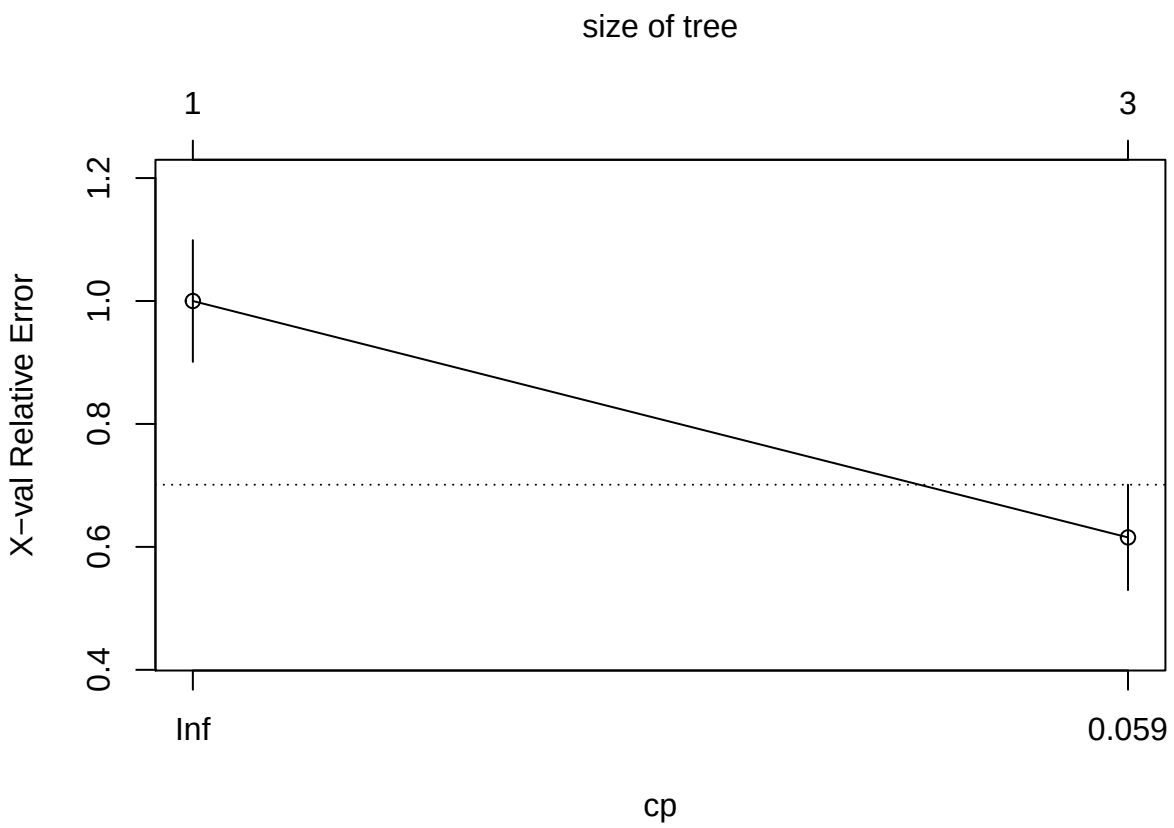



```
printcp(arbol_1)
```

```
##
## Classification tree:
## rpart(formula = renta_media ~ . - Municipios, data = data, method = "class")
##
## Variables actually used in tree construction:
## [1] paro_por_100hab poblacion
##
## Root node error: 65/179 = 0.36313
##
## n= 179
##
##         CP nsplit rel error  xerror     xstd
## 1 0.34615      0   1.00000 1.00000 0.098985
## 2 0.01000      2   0.30769 0.61538 0.085743
```

La evolución del error a medida que se incrementan los nodos se representa mediante la gráfica que aparece debajo.

```
plotcp(arbol_1)
```



Llevamos a cabo la matriz de confusión y la precisión del árbol. Obtenemos una precisión del 88.82%, siendo este el método de mayor precisión.

```
arbolresult <- predict(arbol_1, newdata = data, type = "class") # Predice clasificando entre yes y no

# Matriz de confusi?n
matriz_arbol<-table(arbolresult, data$renta_media)
matriz_arbol
```

```
##
## arbolresult superior inferior
##    superior       51        6
##    inferior       14      108
```

```
# Porcentaje de aciertos
sum(diag(matriz_arbol))/sum(matriz_arbol)
```

```
## [1] 0.8882682
```

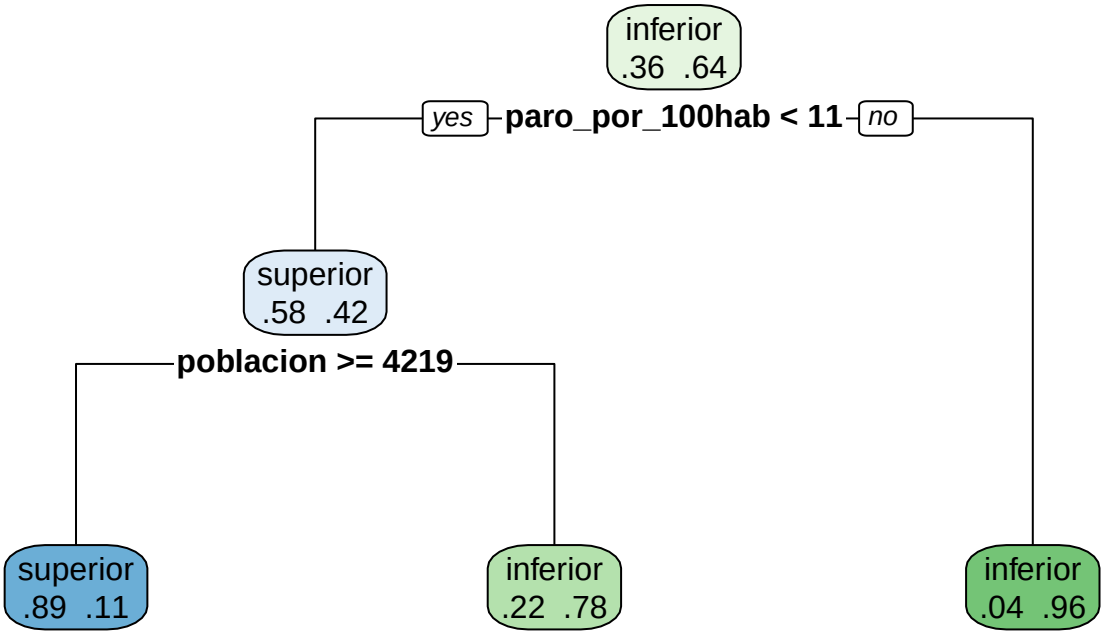
```
rm(t1)
rm(predrpart)
```

Probamos a realizar una poda del árbol, sin embargo vemos que el árbol óptimo es el que se ha mostrado anteriormente.

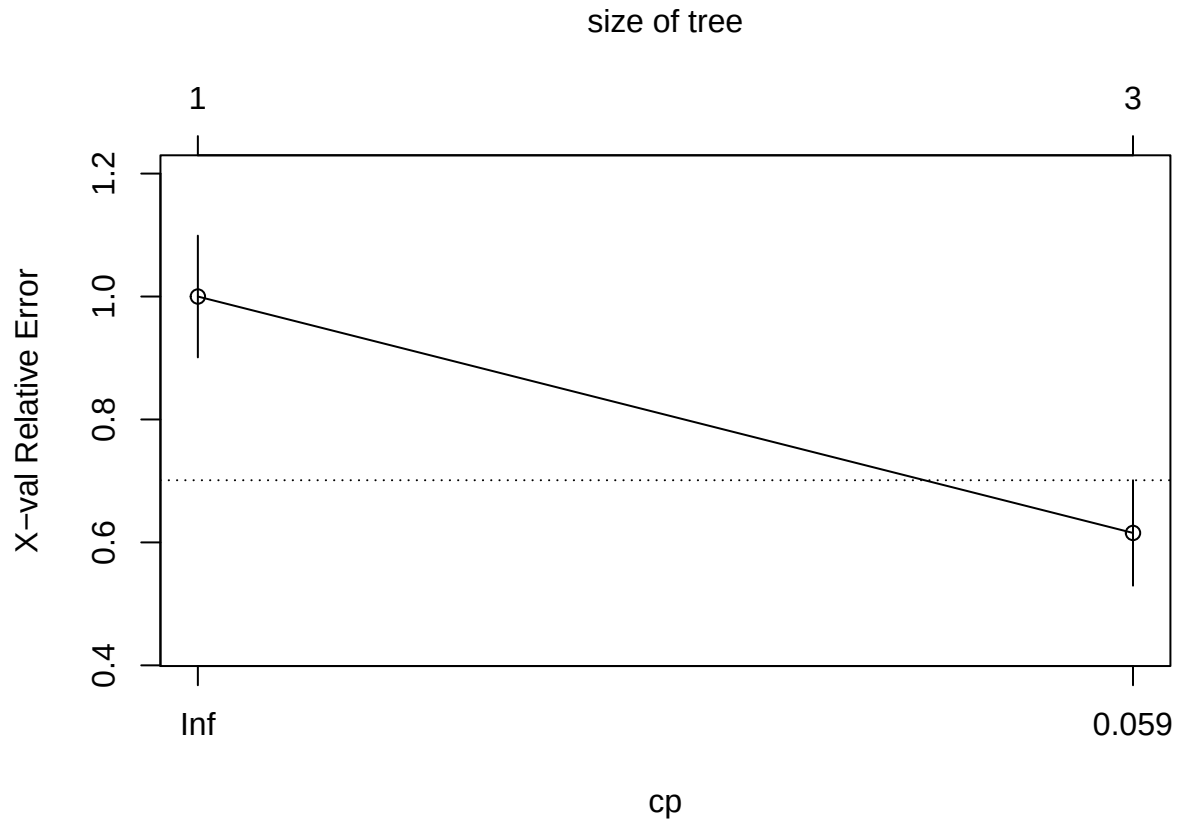
```
optimo_automatic_arboldata_1 <- prune(arbol_1, cp = arbol_1$cptable[which.min(arbol_1$cptable[,"xerror"]

rpart.plot(optimo_automatic_arboldata_1, extra = 4, main = "árbol automático")
```

árbol automático




```
plotcp(optimo_automatic_arboldata_1)
```



Conclusiones:

Dadas las características del dataset, el modelo con mayor precisión, y por tanto, el más adecuado es el modelo de árbol de decisión con un 88.82% de precisión.