A3 Modelización Predictiva

Francisco Javier Melchor González

12/12/2020

Contents

Paquetes

Los paquetes que se van a utilizar para el desarrollo de esta actividad, son los siguientes:

```
if(!require(DataCombine)){
    install.packages("DataCombine")
    library(DataCombine)
}
if(!require(MLmetrics)){
    install.packages("MLmetrics")
    library(MLmetrics)
}
```

1. Datos y Estadística descriptiva

1.1 Lectura de datos

Enunciado: *En primer lugar, leed el fichero de datos y verificad que los tipos de datos se interpretan correctamente. Si fuera necesario, haced las oportunas conversiones de tipos.*

Solución:

En primer lugar, se realiza la lectura del fichero **house.csv**, aplicando para ello la función *read.csv*.

En este caso, se indicarán como parámetros que el dataset sí tiene header *(header=TRUE)*, que el separador de columnas es el ';' *(sep=";")*, que los strings a interpretar como NA son tanto los campos vacíos, los que tienen un espacio en blanco y en los que aparece la cadena "NA" *(na.strings=c(""," ","NA"))* y por último, que las columnas de tipo String, sean consideradas como factores, ya que todas las columnas que son de tipo String, en este caso son factores.

```
house_filepath <- "../Data/house.csv"
house <- read.csv(file=house_filepath, header=TRUE, sep=";", na.strings=c("","  ","NA"),stringsAsFactors
head(house)
```

```
##   price resid_area air_qual room_num   age dist1 dist2 dist3 dist4 teachers
## 1     5      48.10    0.693    5.453 100.0  1.57  1.26  1.79  1.34     19.8
## 2    12      48.10    0.614    5.304  97.3  2.28  1.99  2.41  1.73     19.8
## 3    14      51.89    0.624    6.174  93.6  1.86  1.54  1.87  1.18     18.8
## 4    18      51.89    0.624    6.431  98.8  1.96  1.61  1.92  1.77     18.8
## 5    19      35.19    0.515    5.985  45.4  4.89  4.64  5.05  4.67     19.8
## 6    20      35.96    0.499    5.841  61.4  3.39  3.28  3.62  3.22     20.8
##   poor_prop airport n_hos_beds n_hot_rooms waterbody rainfall bus_ter
## 1     30.59      NO       9.30      13.040      Lake       26     YES
## 2     24.91      NO       9.34      15.096      Lake       39     YES
## 3     24.16      NO       5.68      10.112      Lake       28     YES
## 4     15.39      NO       8.16      14.144      None       41     YES
## 5      9.74      NO       6.38      11.152      Lake       28     YES
## 6     11.41      NO       7.50      15.160      None       39     YES
##        parks Sold
## 1 0.06525315    0
## 2 0.06192155    0
## 3 0.05697699    0
## 4 0.05636501    0
## 5 0.04769962    0
## 6 0.04535682    0
```

```
str(house)
```

```
## 'data.frame':    506 obs. of  19 variables:
##  $ price      : num  5 12 14 18 19 20 20 20 21 21 ...
##  $ resid_area : num  48.1 48.1 51.9 51.9 35.2 ...
##  $ air_qual   : num  0.693 0.614 0.624 0.624 0.515 0.499 0.437 0.489 0.538 0.544 ...
##  $ room_num   : num  5.45 5.3 6.17 6.43 5.99 ...
##  $ age        : num  100 97.3 93.6 98.8 45.4 61.4 74.5 100 87.3 58.8 ...
##  $ dist1      : num  1.57 2.28 1.86 1.96 4.89 3.39 4.33 3.95 4.53 4.07 ...
##  $ dist2      : num  1.26 1.99 1.54 1.61 4.64 3.28 3.72 3.86 3.94 3.86 ...
##  $ dist3      : num  1.79 2.41 1.87 1.92 5.05 3.62 4.26 4.14 4.36 4.24 ...
##  $ dist4      : num  1.34 1.73 1.18 1.77 4.67 3.22 3.9 3.55 4.13 3.84 ...
##  $ teachers   : num  19.8 19.8 18.8 18.8 19.8 20.8 21.3 21.4 19 21.6 ...
##  $ poor_prop  : num  30.59 24.91 24.16 15.39 9.74 ...
##  $ airport    : Factor w/ 2 levels "NO","YES": 1 1 1 1 1 1 1 1 1 1 ...
##  $ n_hos_beds : num  9.3 9.34 5.68 8.16 6.38 ...
##  $ n_hot_rooms: num  13 15.1 10.1 14.1 11.2 ...
##  $ waterbody  : Factor w/ 4 levels "Lake","Lake and River",..: 1 1 1 3 1 3 3 3 4 2 ...
##  $ rainfall   : int  26 39 28 41 28 39 22 60 50 36 ...
##  $ bus_ter    : Factor w/ 1 level "YES": 1 1 1 1 1 1 1 1 1 1 ...
##  $ parks      : num  0.0653 0.0619 0.057 0.0564 0.0477 ...
##  $ Sold       : int  0 0 0 0 0 0 0 0 0 0 ...
```

Como se puede observar en la visualización ofrecida por la función *str(house)*, casi todos los datos han sido captados correctamente, excepto la variable *Sold*, que es interpretada por R como un entero y realmente es de tipo factor, pues indica si la venta ha sido vendida con el 1 y que no ha sido vendida con el 0. A continuación se procede a realizar una conversión de la misma a factor.

```
house$Sold <- as.factor(house$Sold)
str(house)
```

```
## 'data.frame':    506 obs. of  19 variables:
##  $ price      : num  5 12 14 18 19 20 20 20 21 21 ...
##  $ resid_area : num  48.1 48.1 51.9 51.9 35.2 ...
##  $ air_qual   : num  0.693 0.614 0.624 0.624 0.515 0.499 0.437 0.489 0.538 0.544 ...
##  $ room_num   : num  5.45 5.3 6.17 6.43 5.99 ...
##  $ age        : num  100 97.3 93.6 98.8 45.4 61.4 74.5 100 87.3 58.8 ...
##  $ dist1      : num  1.57 2.28 1.86 1.96 4.89 3.39 4.33 3.95 4.53 4.07 ...
##  $ dist2      : num  1.26 1.99 1.54 1.61 4.64 3.28 3.72 3.86 3.94 3.86 ...
##  $ dist3      : num  1.79 2.41 1.87 1.92 5.05 3.62 4.26 4.14 4.36 4.24 ...
##  $ dist4      : num  1.34 1.73 1.18 1.77 4.67 3.22 3.9 3.55 4.13 3.84 ...
##  $ teachers   : num  19.8 19.8 18.8 18.8 19.8 20.8 21.3 21.4 19 21.6 ...
##  $ poor_prop  : num  30.59 24.91 24.16 15.39 9.74 ...
##  $ airport    : Factor w/ 2 levels "NO","YES": 1 1 1 1 1 1 1 1 1 1 ...
##  $ n_hos_beds : num  9.3 9.34 5.68 8.16 6.38 ...
##  $ n_hot_rooms: num  13 15.1 10.1 14.1 11.2 ...
##  $ waterbody  : Factor w/ 4 levels "Lake","Lake and River",..: 1 1 1 3 1 3 3 3 4 2 ...
##  $ rainfall   : int  26 39 28 41 28 39 22 60 50 36 ...
##  $ bus_ter    : Factor w/ 1 level "YES": 1 1 1 1 1 1 1 1 1 1 ...
##  $ parks      : num  0.0653 0.0619 0.057 0.0564 0.0477 ...
##  $ Sold       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

Una vez que todas las variables tienen sus tipos asignados, se procede a continuación a ver la calidad y la distribución de los datos que forman el dataframe a analizar.

1.2 Descriptiva y visualización

Enunciado: *A continuación, comenzaremos el estudio descriptivo, para caracterizar el tipo de variables, detectar posible datos faltantes, outliers, variables con varianza nula o casi nula, etc.*

Solución:

En primer lugar, se procederá a comprobar si existen datos faltantes en el dataset a analizar.

```
colSums(is.na(house))
```

```
##       price  resid_area    air_qual    room_num         age       dist1
##           0           0           0           0           0           0
##       dist2       dist3       dist4    teachers   poor_prop     airport
##           0           0           0           0           0           0
##  n_hos_beds n_hot_rooms   waterbody    rainfall     bus_ter       parks
##           8           0           0           0           0           0
##        Sold
##           0
```

Como se puede observar, existen 8 valores faltantes correspondientes a la columna n_hos_beds. Al tratarse de un número tan pequeño con respecto al total de datos, se procede a eliminar directamente todas aquellas filas que contengan valores faltantes.

```
house = DropNA(house)
```

```
## No Var specified. Dropping all NAs from the data frame.
```

```
## 8 rows dropped from the data frame because of missing values.
```

Una vez comprobada y solventada la existencia de datos faltantes en el dataframe, se procede a visualizar las distintas variables que lo forman.

1.2.1 Representación gráfica de variables categóricas o cualitativas

En primer lugar, se realizará la representación gráfica de aquellas variables categóricas o cualitativas, estas son las siguientes:

```
factors = unlist(lapply(house, is.factor))
which(factors, arr.ind = TRUE)
```

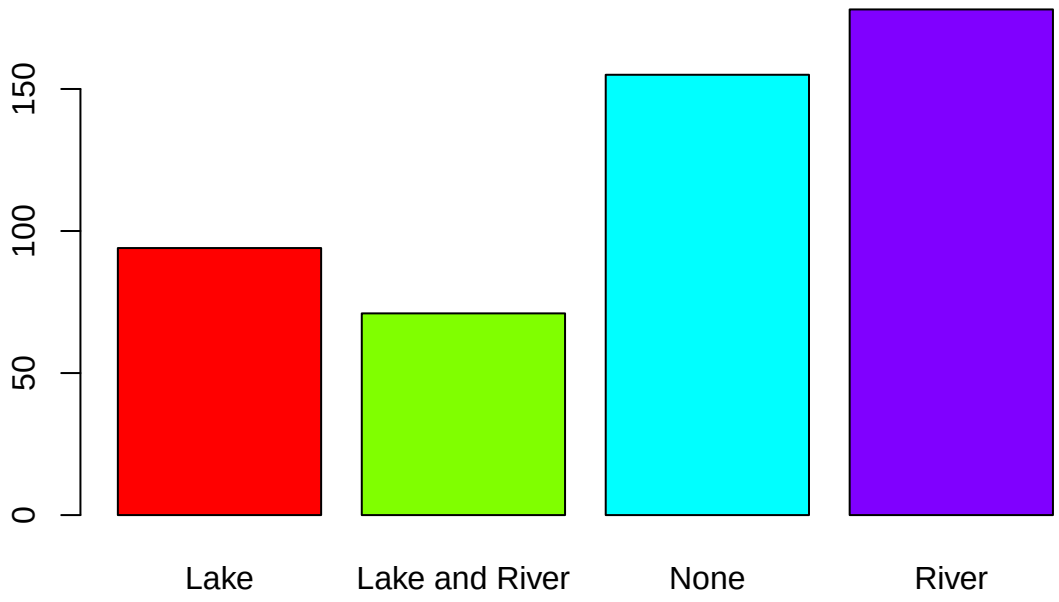
```
##   airport waterbody   bus_ter      Sold
##        12        15        17        19
```

Como se puede observar, las únicas variables cualitativas son:

- **airport**, que indica si hay un aeropuerto o no en la zona donde se ubica la vivienda
- **waterbody**, que indica el tipo de fuente natural de agua dulce hay en la ciudad donde se encuentra la vivienda a analizar
- **bus_ter**, que indica si hay, al menos, una terminal de buses en la ciudad
- **Sold**, que indica si la propiedad se vendió (1) o no (0)

A continuación se porcede a representar las mismas

```
counts <- table(house$waterbody)
barplot(counts, main="Distribución de tipos de fuente natural de agua dulce
        que hay en la ciudad", xlab="Número de fuentes por cada categoría",
        cex.main = 0.8, cex.lab = 0.8,
        col = rainbow (length(levels(house$waterbody))))
```



Número de fuentes por cada categoría

```
colorForPieCharts = rainbow(length(levels(house$airport)) +
                            length(levels(house$bus_ter)) +
                            length(levels(house$Sold)))
par(mfrow=c(1,3))

levels(house$airport)
```

```
## [1] "NO"  "YES"
```

```
mytableAirport <- table(house$airport)
pctAirport <- round(mytableAirport/sum(mytableAirport)*100)
lblsAirport <- paste(names(mytableAirport), "\n", pctAirport, sep="")
lblsAirport <- paste (lblsAirport, '%', sep="")
pie(mytableAirport, labels = lblsAirport, col=colorForPieCharts[1:2],
    main="Pie Chart of Airport\n",cex.main = 0.8)


levels(house$bus_ter)
```

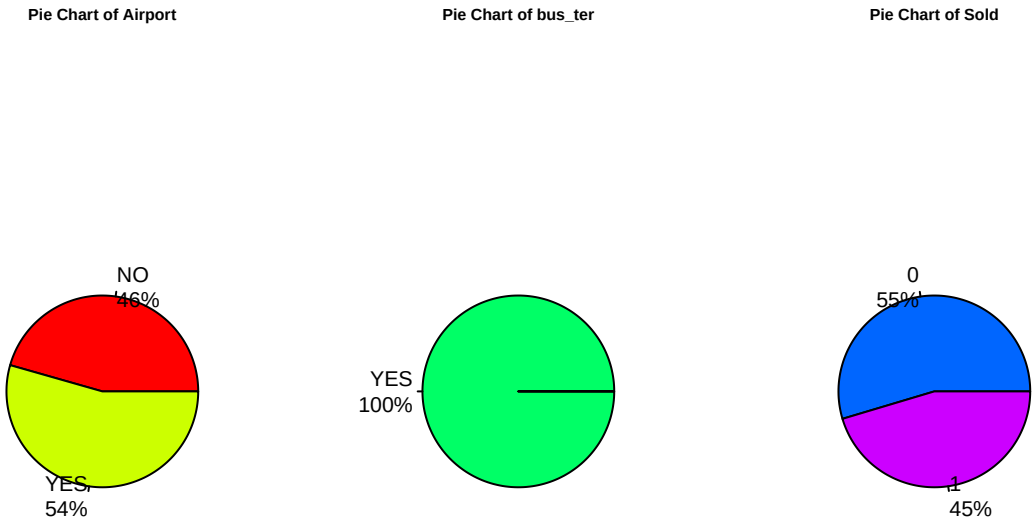
```
## [1] "YES"
```

```
mytableBus_ter <- table(house$bus_ter)
pctBus_ter <- round(mytableBus_ter/sum(mytableBus_ter)*100)
lblsBus_ter <- paste(names(mytableBus_ter), "\n", pctBus_ter, sep="")
lblsBus_ter <- paste (lblsBus_ter, '%', sep="")
pie(mytableBus_ter, labels = lblsBus_ter, col=colorForPieCharts[3:3],
    main="Pie Chart of bus_ter\n",cex.main = 0.8)
```

```
levels(house$Sold)
```

```
## [1] "0" "1"
```

```
mytableSold <- table(house$Sold)
pctSold <- round(mytableSold/sum(mytableSold)*100)
lblsSold <- paste(names(mytableSold), "\n", pctSold, sep="")
lblsSold <- paste (lblsSold, '%', sep="")
pie(mytableSold, labels = lblsSold, col=colorForPieCharts[4:5],
    main="Pie Chart of Sold\n",cex.main = 0.8)
```



De estas visualizaciones, cabe destacar la variable *bus_ter*, ya que como se puede ver solo toma un único valor, que es "Yes".

Las otras gráficas no indican nada relevante a destacar, más que la distribución de las variables a las que representan.

1.2.2 Representación gráfica de variables numéricas

En segundo lugar, se realizará la representación gráfica de aquellas variables numéricas, estas son las siguientes:

```
numeric = unlist(lapply(house, is.numeric))
which(numeric, arr.ind = TRUE)
```

```
##       price  resid_area    air_qual    room_num         age       dist1
##           1           2           3           4           5           6
##       dist2       dist3       dist4    teachers   poor_prop  n_hos_beds
##           7           8           9          10          11          13
```

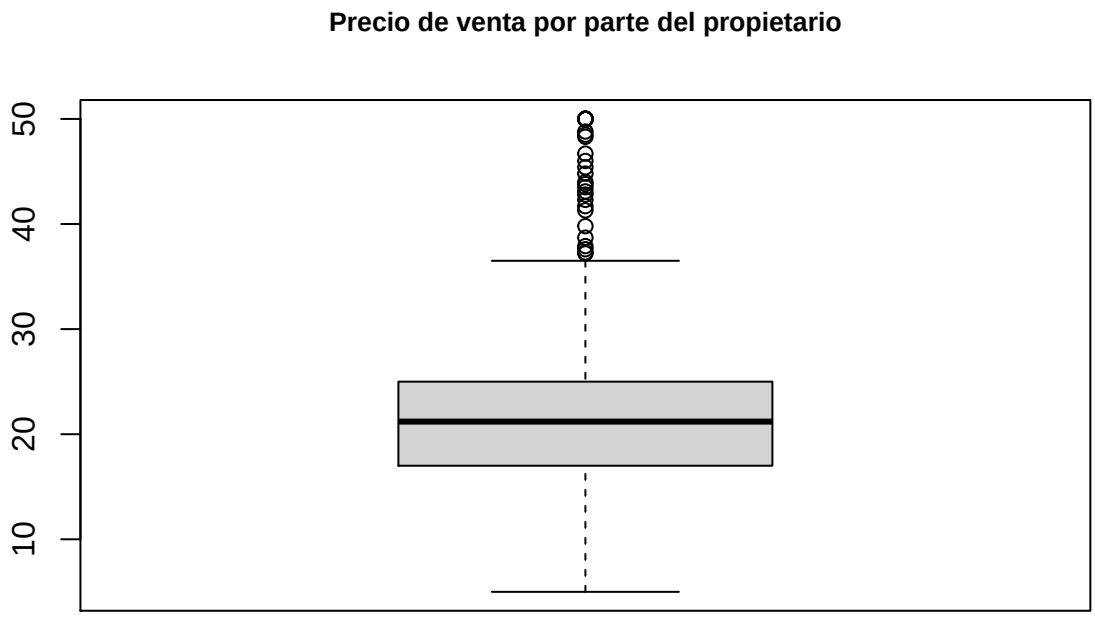
```
## n_hot_rooms    rainfall       parks
##          14          16          18
```

Como se puede observar, las variables numéricas son:

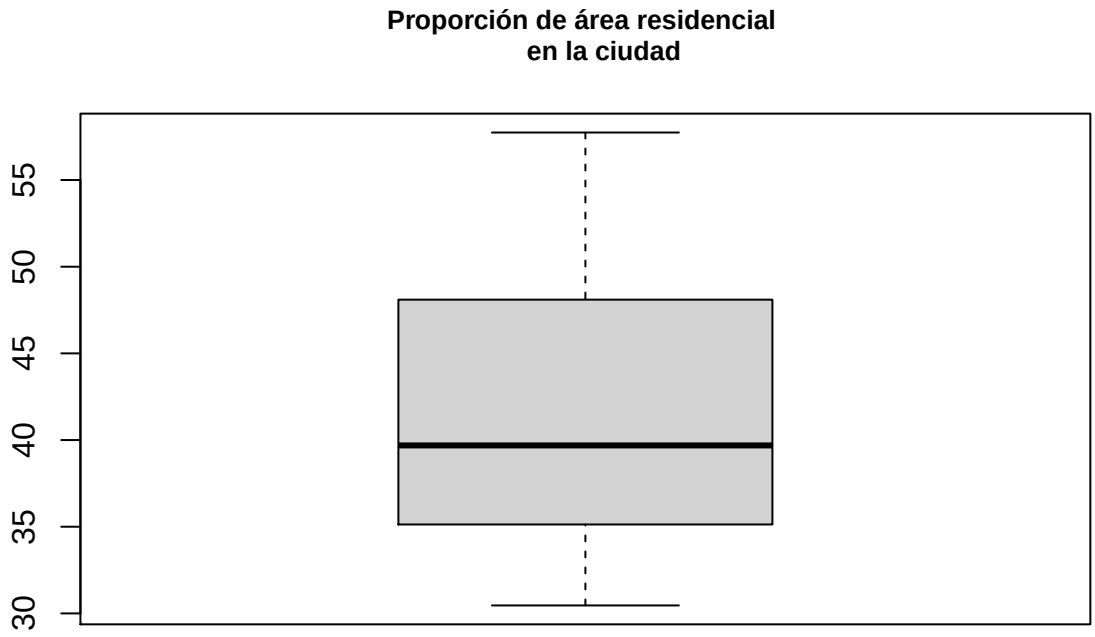
- price, que indica el precio de la vivienda en concreto
- resid_area, que indica la proporción de área residencial en la ciudad
- air_qual, que indica la calidad del aire del vecindario donde se encuentra la vivienda
- room_num, que indica el número medio de habitaciones en casas de esa localidad
- age, que indica los años de construcción inmobiliaria de la vivienda
- dist1, que indica la distancia al centro de empleo 1
- dist2, que indica la distancia al centro de empleo 2
- dist3, que indica la distancia al centro de empleo 3
- dist4, que indica la distancia al centro de empleo 4
- teachers, que indica el número de maestros por cada mil habitantes en el municipio donde se encuentra la vivienda
- poor_prop, que indica la proporción de población pobre de la ciudad donde se encuentra la vivienda
- n_hos_beds, que inidica el número de camas de hospital por mil habitantes en la ciudad donde se encuentra la vivienda
- n_hot_rooms, que indica el número de habitaciones de hotel por cada mil habitantes en la ciudad donde se encuentra la vivienda
- rainfall, que indica la precipitación media anual en centímetros
- parks, que indica la porporción de terrenos asignados como parques y áreas verdes en la ciudad.

A continuación, se representarán las mismas con un diagrama de cajas y bigotes para estudiar la existencia de valores atípicos en las mismas.

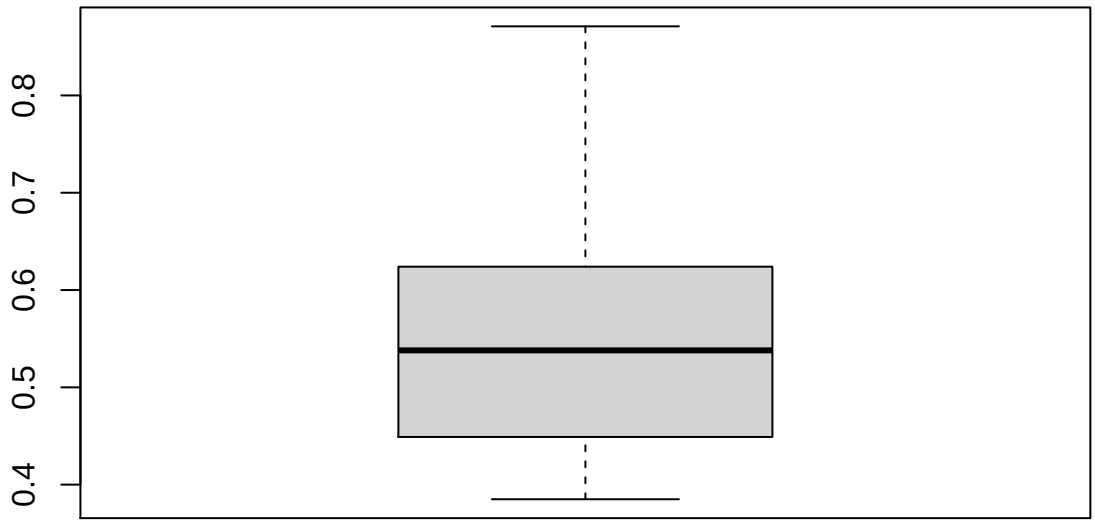
```
boxplot(house$price,main="Precio de venta por parte del propietario",cex.main=0.8)
```



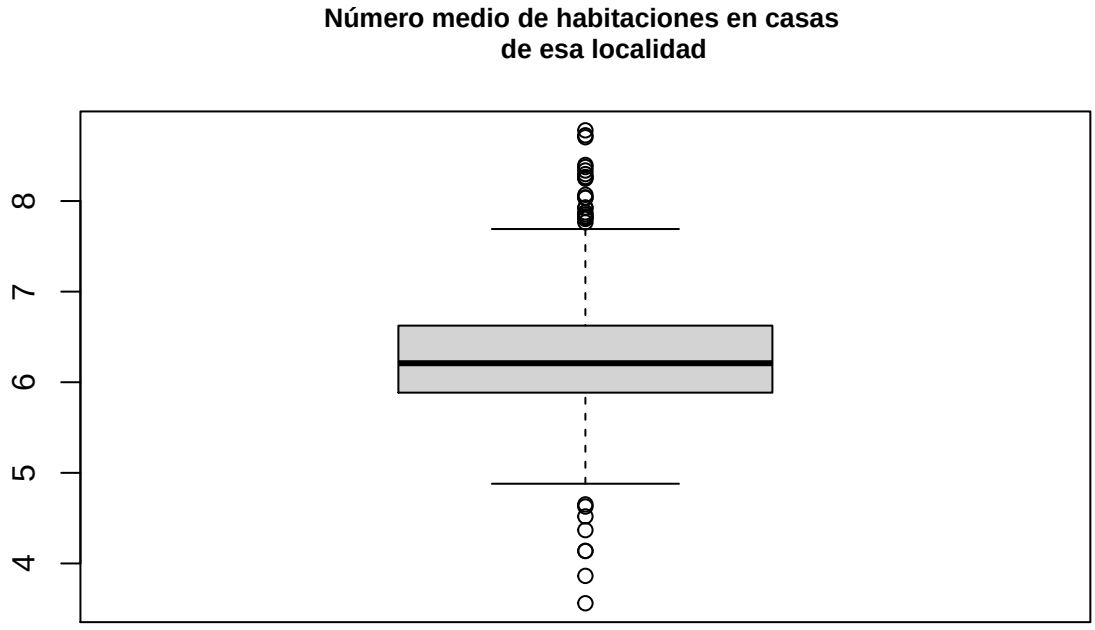
```r
boxplot(house$resid_area,main="Proporción de área residencial
     en la ciudad",cex.main=0.8)
```



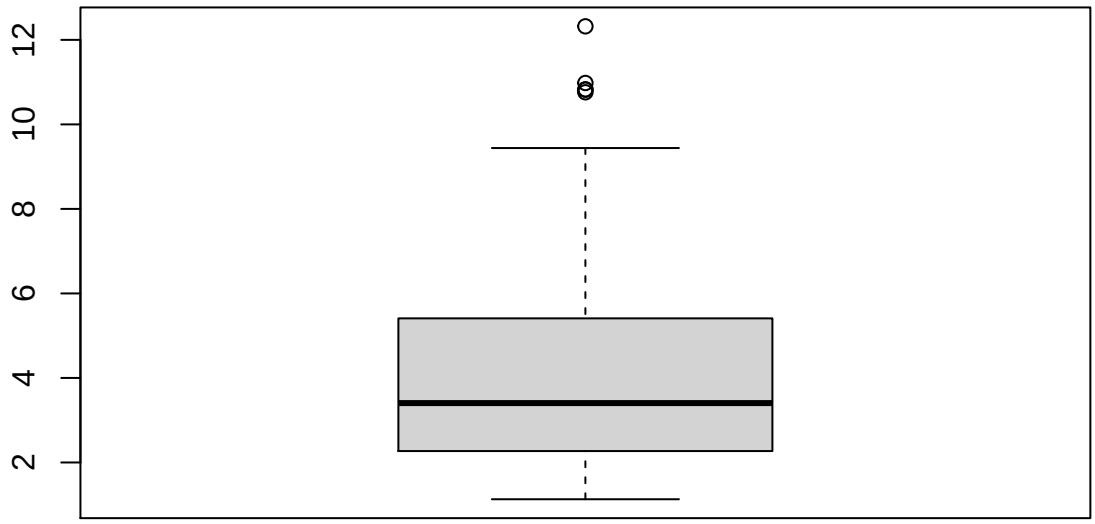

```r
boxplot(house$air_qual,main="Calidad del aire del vecindario",cex.main=0.8)
```



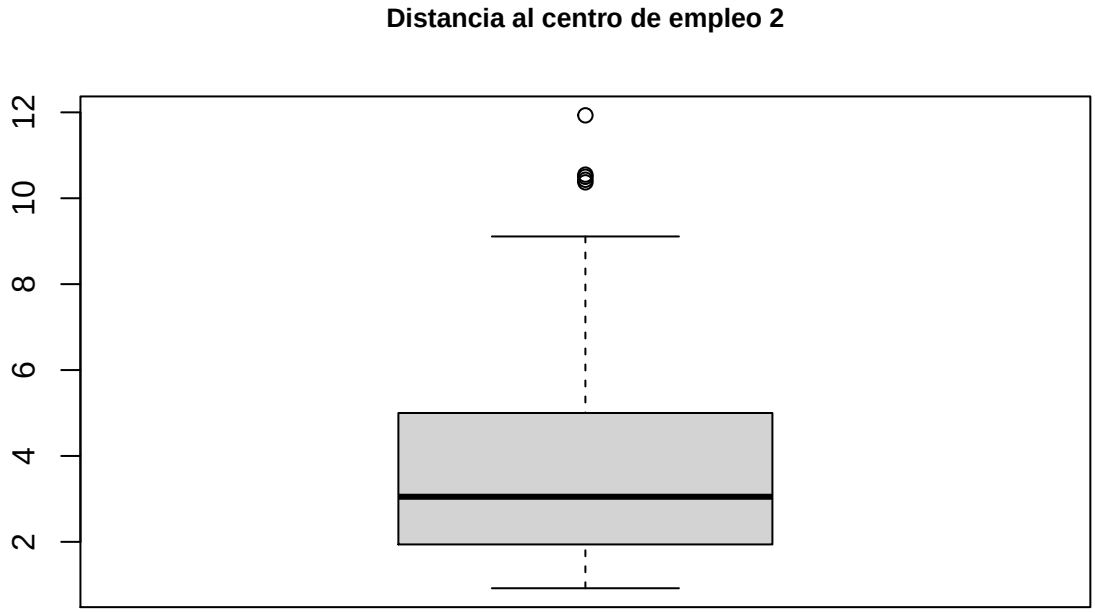

```
boxplot(house$room_num, main="Número medio de habitaciones en casas
     de esa localidad", cex.main=0.8)
```



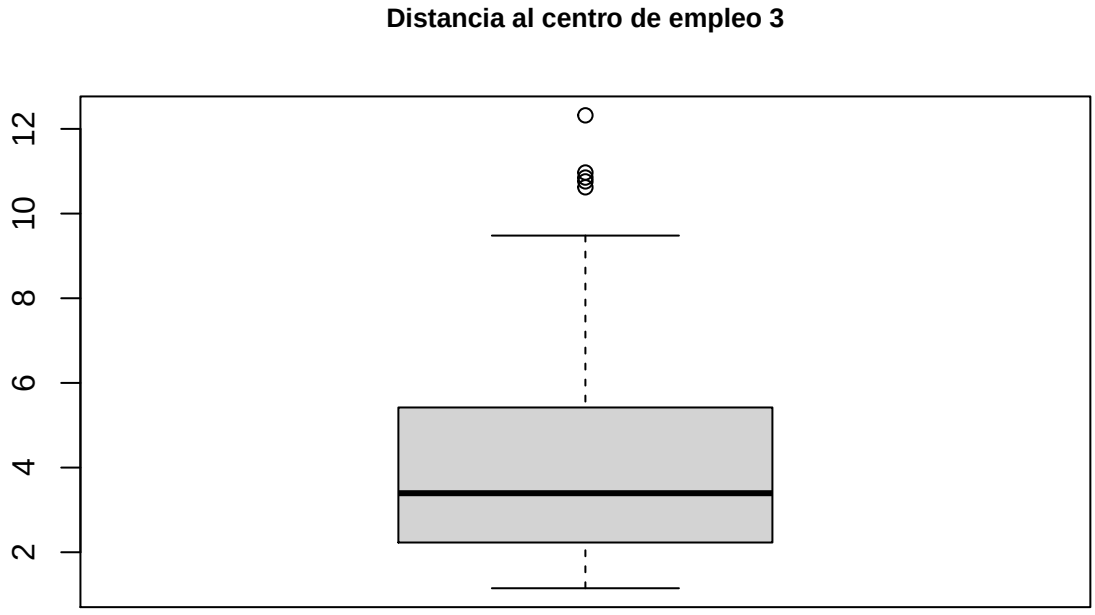

```
boxplot(house$dist1,main="Distancia al centro de empleo 1",
     cex.main=0.8)
```



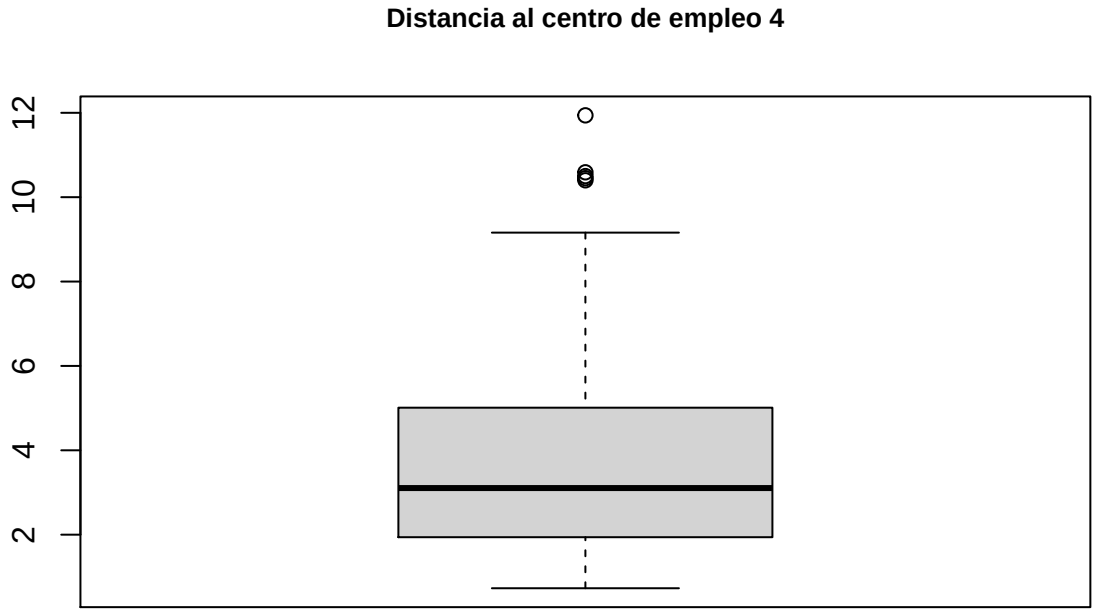

```
boxplot(house$dist2, main="Distancia al centro de empleo 2",
     cex.main=0.8)
```



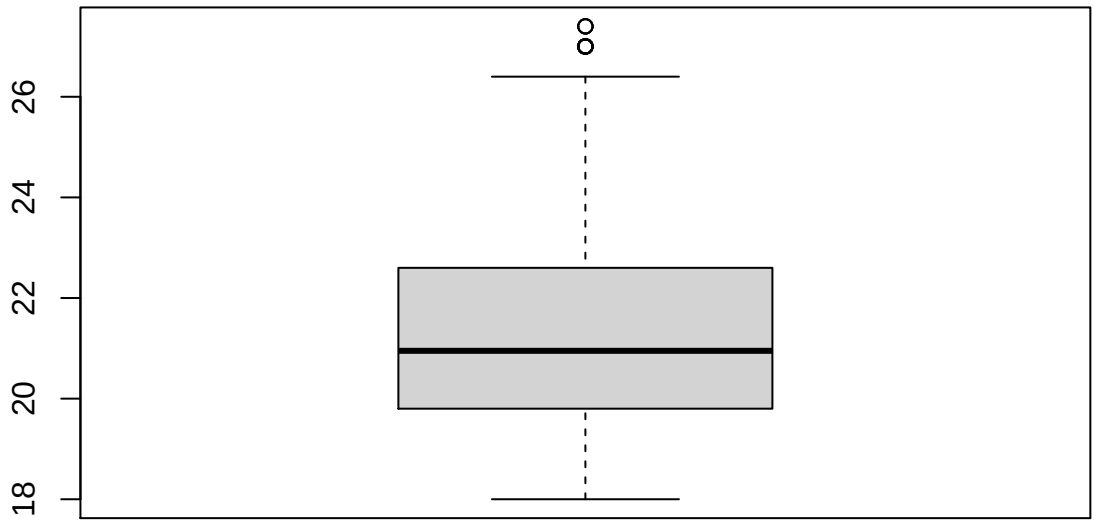

```
boxplot(house$dist3,main="Distancia al centro de empleo 3",
     cex.main=0.8)
```



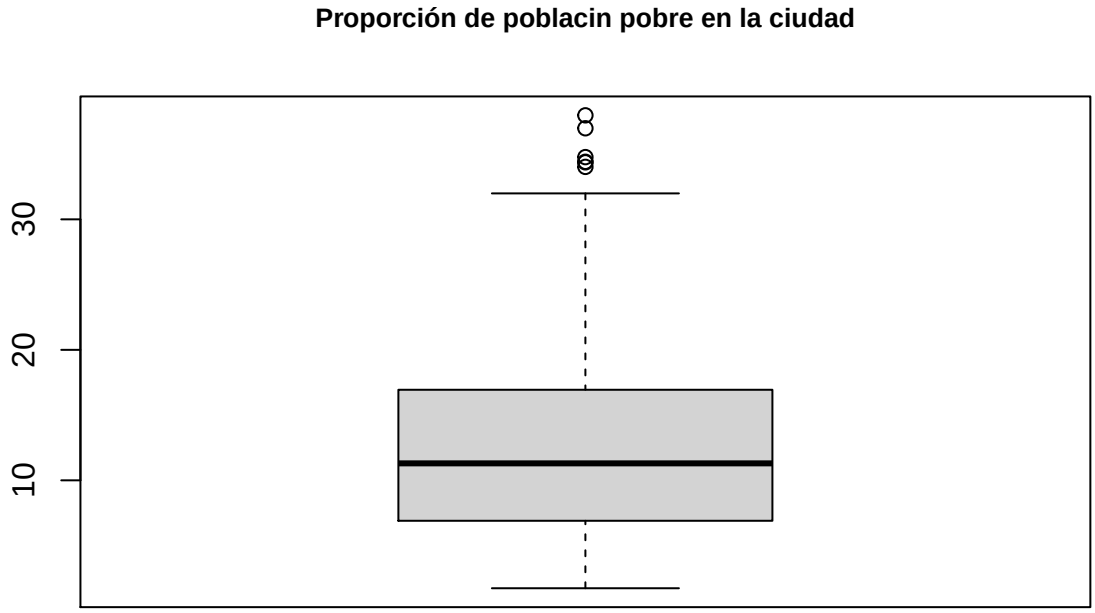

```
boxplot(house$dist4,main="Distancia al centro de empleo 4",
     cex.main=0.8)
```



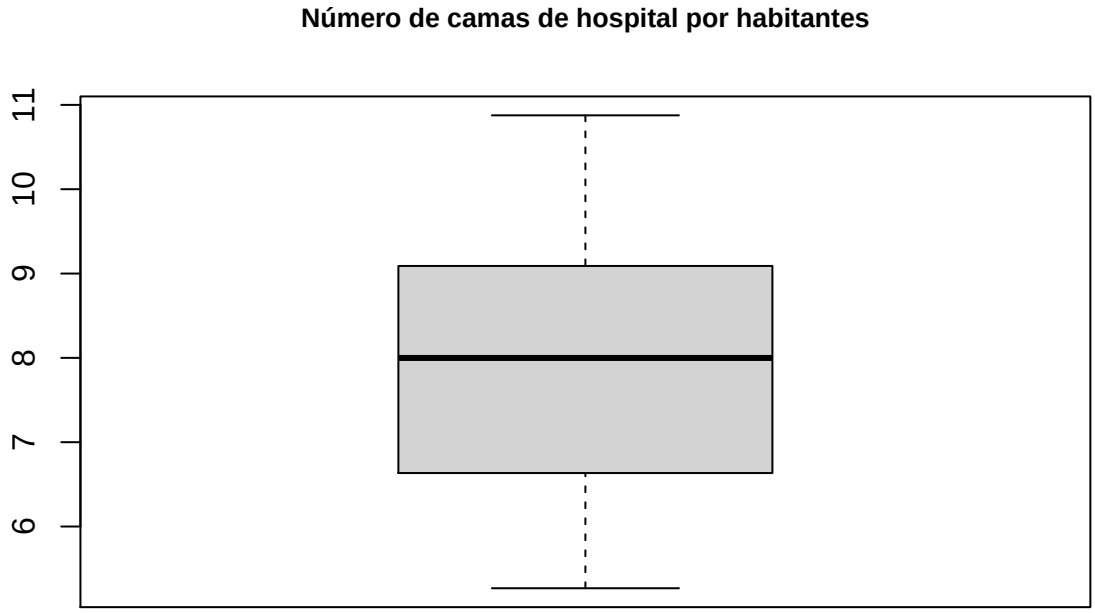

```
boxplot(house$teachers,main="Número de maestros en el municipio",
     cex.main=0.8)
```



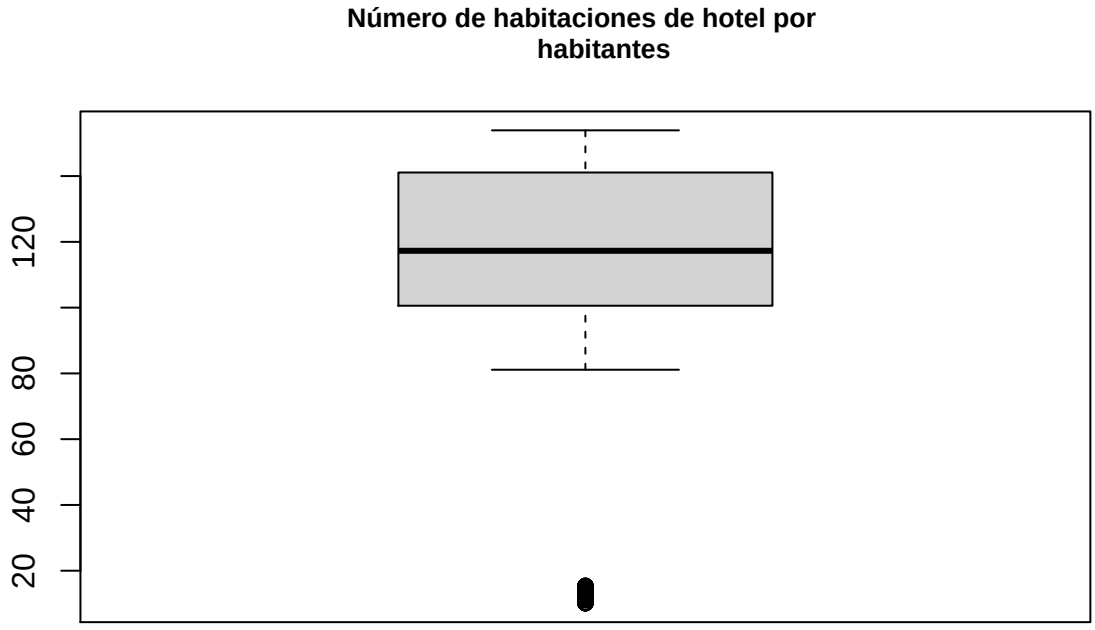

```
boxplot(house$poor_prop, main="Proporción de poblacin pobre en la ciudad",
     cex.main=0.8)
```



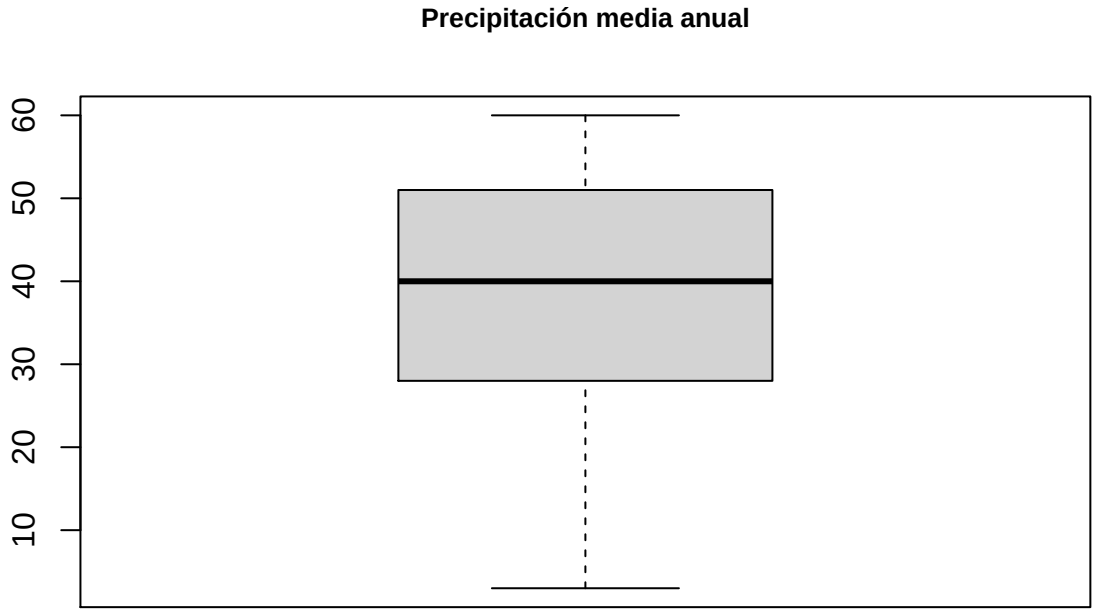

```
boxplot(house$n_hos_beds, main="Número de camas de hospital por habitantes",
     cex.main=0.8)
```



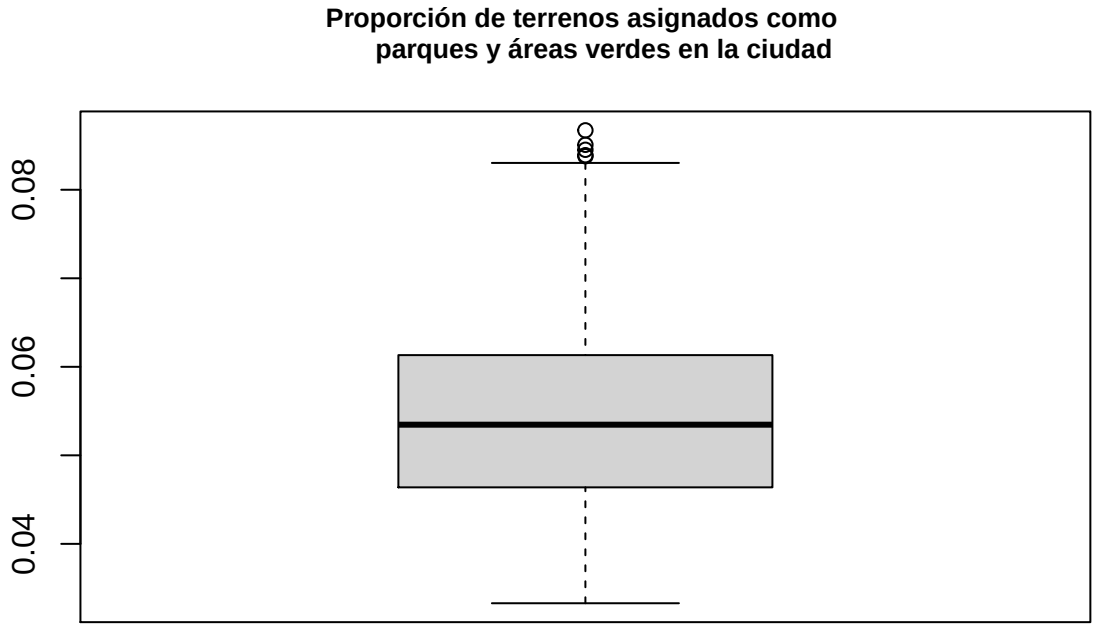

```
boxplot(house$n_hot_rooms, main="Número de habitaciones de hotel por
    habitantes", cex.main=0.8)
```




```
boxplot(house$rainfall, main="Precipitación media anual",
     cex.main=0.8)
```




```
boxplot(house$parks, main="Proporción de terrenos asignados como
     parques y áreas verdes en la ciudad",
     cex.main=0.8)
```



Como se puede observar, existen valores atípicos en la gran mayoría de variables numéricas, estas son:

- price
- room_num
- dist1
- dist2
- dist3
- dist4
- teachers
- poor_prop
- n_hot_rooms
- parks

Esto hará que algunos modelos estén sesgados por dichos valores y que estos no realicen correctamente las predicciones.

2. Modelo de regresión lineal

2.1. Modelo de regresión lineal simple

2.1.1. Calcular

Enunciado:

Estimar por mínimos cuadrados ordinarios dos modelos lineales que expliquen la variable price, uno en función de la variable teachers y otro en función de la variable poor_prop.

Solución

Para realizar la estimación por mínimos cuadrados ordinarios de los diferentes modelos lineales desarrollados en esta práctica, se han desarrollado las siguientes funciones:

```
get_cov_muestral<- function(x,y){
    mean_x = mean(x)
    mean_y = mean(y)
    sum = 0
    for (i in 1:length(x)){
        sum = sum + ((x[i] - mean_x)*(y[i] - mean_y))
    }
    return (sum/(length(x) - 1))
}

get_var_muestral <- function(x){
    mean_x = mean(x)
    sum = 0
    for (i in 1:length(x)){
        sum = sum + ((x[i]-mean_x)^2)
    }
    return (sum/(length(x) - 1))
}

get_b1 <- function(x,y){
    Sxy = get_cov_muestral(x,y)
    S2x = get_var_muestral(x)

    return (Sxy/S2x)
}

get_b0 <- function(x,y){
    mean_y = mean(y)
    b1 = get_b1(x,y)
    mean_x = mean(x)

    return(mean_y - (b1*mean_x))
}
```

Se procede a continuación a realizar la estimación para los modelos indicados en el enunciado:

```
b0_teachers = get_b0(house$teachers,house$price)
b1_teachers = get_b1(house$teachers,house$price)

b0_poor_prop = get_b0(house$poor_prop,house$price)
b1_poor_prop = get_b1(house$poor_prop,house$price)
```

Como se puede observar, se obtienen los siguientes modelos:

- teachers= -23.475578+ 2.1379667x
- poor_prop= 34.5467435 -0.9499102x

Para demostrar que los diferentes modelos han sido calculados correctamente, se estiman los mismos a través de la función lm de R:

```
lm(price ~ teachers, data =house)
```

```
##
## Call:
```

```
## lm(formula = price ~ teachers, data = house)
##
## Coefficients:
## (Intercept)     teachers
##     -23.476        2.138
```

```
lm(price ~ poor_prop, data=house)
```

```
##
## Call:
## lm(formula = price ~ poor_prop, data = house)
##
## Coefficients:
## (Intercept)    poor_prop
##     34.5467      -0.9499
```

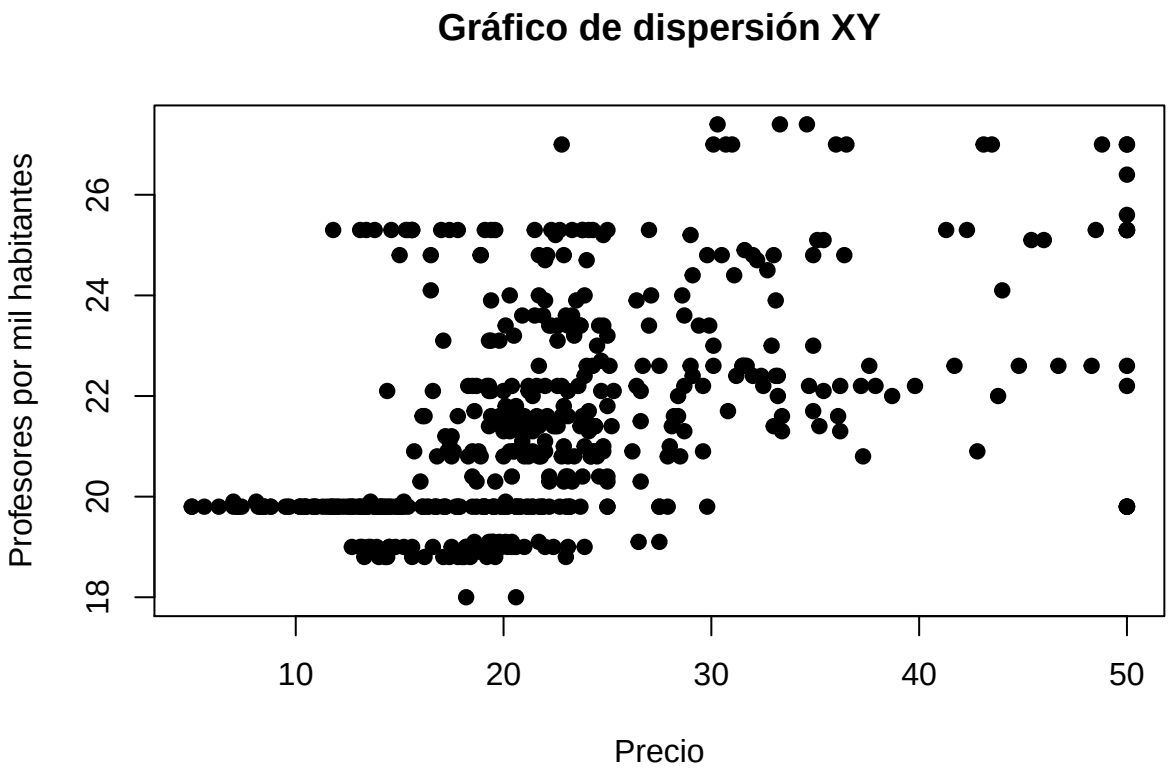
2.1.2. Describe las diferencias entre ambos modelos y compáralos.

Las diferencias que existen entre ambos modelos son las siguientes:

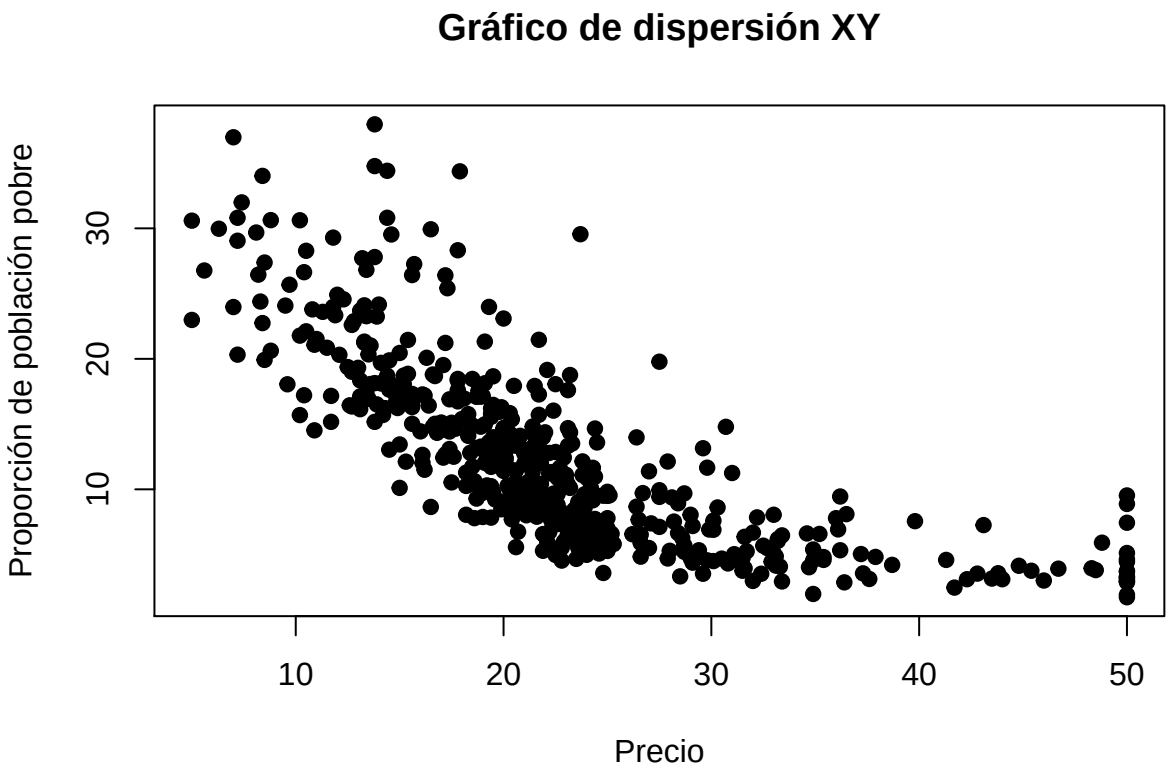
- 1. El primer modelo presenta una pendiente positiva mientras que el segundo presenta una pendiente negativa, esto es debido a que la variable *teachers* influye de manera creciente en el precio de la vivienda y que la variable *poor_prop*, por el contrario, influye de manera decreciente en el precio de la vivienda.

2.1.3. Para cada modelo, realiza un gráfico de dispersión XY e interpretar brevemente el gráfico resultante.

```
plot(house$price, house$teachers, main="Gráfico de dispersión XY",
     ylab="Profesores por mil habitantes", xlab="Precio", pch=19)
```




```
plot(house$price, house$poor_prop, main="Gráfico de dispersión XY",
     ylab="Proporción de población pobre", xlab="Precio", pch=19)
```



A través de las gráficas anteriores, se puede observar que la variable *teachers* está relacionada con el precio pero en menor manera que la variable *poor_prop*, pues la primera gráfica realmente no presenta una clara subida o bajada del precio, sino que este se encuentra más repartido entre los diferentes valores de *teachers*, mientras que en la segunda gráfica, se aprecia perfectamente como a medida que disminuye la proporción de profesores en la población de la ciudad donde se encuentra la vivienda, disminuye también el precio de la misma.

Realmente, lo que reflejan las gráficas, tiene sentido, pues los profesores, normalmente, suelen tener una estabilidad económica mayor que otros tipos de trabajos, por lo que se podrán permitir casas un poco más caras que la gente de clase más obrera.

2.2. Modelo de regresión lineal múltiple (regresores cuantitativos)

2.2.1. Calcular

Enunciado:

Estimar por mínimos cuadrados ordinarios un modelo lineal que explique la variable price en función de age, teachers, poor_prop

Solución: A continuación, se procede a estimar por mínimos cuadrados el modelo de regresión lineal múltiple indicado en el enunciado

```
get_b <- function(X,y){
    first_term = solve (t(X) %*% X)
    sec_term = t(X) %*% y
    return (first_term %*% sec_term)
}
X = cbind(rep(1,length(house$price)),house$age,house$teachers,house$poor_prop)
```

```
y = house$price
B = get_b(X,y)
B
```

```
##              [,1]
## [1,]  6.68237473
## [2,]  0.04018712
## [3,]  1.14617873
## [4,] -0.91633870
```

```
B0 = B[1]
B1 = B[2]
B2 = B[3]
B3 = B[4]
```

Para comprobar que los cálculos se realizan de forma correcta, se procede a continuación a calcular el modelo a través de la función lm de R:

```
lm1 = lm(price ~ age+teachers+poor_prop, data = house)
```

2.2.2. Indicar el efecto de cada variable regresora e interpretar el modelo.

A través de los diferentes coeficientes del modelo anterior calculado, se puede observar que el efecto de las diferentes variables regresoras es el siguiente:

- En el caso de la variable age, el coeficiente 0.04 indica que por cada año de antigüedad de la vivienda, esta aumentará 0.04€ su valor total.
- En el caso de la variable teachers, el coeficiente 1.146 indica que por cada profesor que haya por cada mil habitantes, el precio de la vivienda aumentará 1.146€.
- En el caso de la variable poor_prop, el coeficiente -0.913 indica que a medida que aumente en una unidad la proporción de población pobre, el valor de la vivienda decrecerá 0.913€

2.2.3. Evaluar la bondad de ajuste a través del coeficiente de determinación ajustado.

Se procede a continuación a evaluar la bondad del ajuste realizado por el modelo utilizando para ello un conjunto de funciones creadas a continuación:

```
rlm_1 <- function(age,teachers,poor_prop){
    return (B0 + B1*age + B2*teachers + B3*poor_prop)
}
predict_rlm_1 <- function(age,teachers,poor_prop){
    y <- c()
    for (i in 1:length(age)){
        y[i] <- rlm_1(age[i],teachers[i],poor_prop[i])
    }
    return(y)
}
get_sct <- function(y){
    D = y - mean(y)
    return (t(D) %*% D)
}

get_scr <- function(y, y_predict){
    W = y_predict - mean(y)
    return (t(W) %*% W)
```

```
}

get_r_square <- function (y, y_predict){
    return (get_scr(y, y_predict) / get_sct(y))
}

y_predict_rlm1 = predict_rlm_1(house$age,house$teachers,house$poor_prop)
get_r_square(y, y_predict_rlm1)
```

```
##           [,1]
## [1,] 0.6161525
```

Para comprobar que el cálculo se realiza correctamente por las funciones, se procede a continuación a calcular el coeficiente de determinación a través de las funciones que ofrece R:

```
summary(lm1)$r.squared
```

```
## [1] 0.6161525
```

Como se puede observar, el valor obtenido es de 0.61, lo que indica que el modelo de regresión múltiple calculado explica el 61% de la variabilidad del precio. No se trata de un mal resultado, ya que se encuentra por encima de del 50%, pero queda un 39% de variabilidad por explicar, lo que indica que no es un mal modelo pero que se podría mejorar.

2.2.4. Ampliar el modelo anterior con las variables room_num, n_hos_beds y n_hot_rooms.

Enunciado:

Comparar los dos modelos. ¿Es significativamente mejor el nuevo modelo?

```
X = cbind(rep(1,length(house$price)),house$age,house$teachers,house$poor_prop, house$room_num, house$n_h
          house$n_hot_rooms)
B = get_b(X,y)
B
```

```
##              [,1]
## [1,] -19.93832398
## [2,]   0.01194388
## [3,]   0.91717946
## [4,]  -0.59469971
## [5,]   4.38650028
## [6,]   0.39513139
## [7,]  -0.01282008
```

```
B0 = B[1]
B1 = B[2]
B2 = B[3]
B3 = B[4]
B4 = B[5]
B5 = B[6]
B6 = B[7]

rlm_2 <- function(age,teachers,poor_prop,room_num,n_hos_beds,n_hot_rooms){
    return (B0 + B1*age + B2*teachers + B3*poor_prop + B4*room_num + B5*n_hos_beds + B6*n_hot_rooms)
}
predict_rlm_2 <- function(age,teachers,poor_prop,room_num,n_hos_beds,n_hot_rooms){
    y <- c()
    for (i in 1:length(age)){
```

```
        y[i] <- rlm_2(age[i],teachers[i],poor_prop[i],room_num[i],n_hos_beds[i],n_hot_rooms[i])
    }
    return(y)
}

y_predict_rlm2 = predict_rlm_2(house$age,house$teachers,house$poor_prop,house$room_num,house$n_hos_beds
get_r_square(y, y_predict_rlm2)
```

```
##           [,1]
## [1,] 0.6915732
```

Como se puede observar, el valor obtenido por el coeficiente de determinación en este caso es de 0.691, lo que indica que hemos logrado mejorar el modelo anterior, pues este consigue explicar un 69% la variabilidad del precio de la vivienda, mientras que el anterior solo era capaz de explicar un 61%. Se podría decir por tanto, que el modelo es mejor que el anterior, pero tampoco lo mejora en gran cantidad.

2.3. Modelo de regresión lineal múltiple (regresores cuantitativos y cualitativos)

Enunciado:

Queremos conocer en qué medida el modelo anterior (Modelo 2.2) se ve afectado por la inclusión de la variable airport

2.3.1. Aplicar un modelo de regresión lineal múltiple y explicar el resultado.

Lo primero que se realizará, será una conversión de la variable airport para poder incluirla en el modelo, pues esta toma valores "YES" y "NO", y los modelos solo aceptan valores numéricos, por lo que asignaremos el valor 0 para el caso del "NO" y 1 en el caso de "YES"

```
head(house$airport)
```

```
## [1] NO NO NO NO NO NO
## Levels: NO YES
```

```
tail(house$airport)
```

```
## [1] YES YES YES YES YES YES
## Levels: NO YES
```

```
levels(house$airport)[levels(house$airport)=="NO"] <- 0
levels(house$airport)[levels(house$airport)=="YES"] <- 1
head(house$airport)
```

```
## [1] 0 0 0 0 0 0
## Levels: 0 1
```

```
tail(house$airport)
```

```
## [1] 1 1 1 1 1 1
## Levels: 0 1
```

```
house$airport <- as.integer(as.character(house$airport))
head(house$airport)
```

```
## [1] 0 0 0 0 0 0
```

```
tail(house$airport)
```

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

Una vez convertida la variable, se procede a aplicar el modelo solicitado en el enunciado:

```
X = cbind(rep(1,length(house$price)),house$age,house$teachers,house$poor_prop, house$room_num, house$n_
          house$n_hot_rooms,house$airport)
B = get_b(X,y)
B
```

```
##              [,1]
## [1,] -19.98178102
## [2,]   0.01164043
## [3,]   0.91694608
## [4,]  -0.59702182
## [5,]   4.44100479
## [6,]   0.40261060
## [7,]  -0.01172600
## [8,]  -0.75586920
```

```
B0 = B[1]
B1 = B[2]
B2 = B[3]
B3 = B[4]
B4 = B[5]
B5 = B[6]
B6 = B[7]
B7 = B[8]
```

A través del modelo anterior se pueden extraer las siguientes conclusiones:

- Cuando todas las variables se encuentren a 0, el valor de la vivienda será igual a -19.98, esto realmente no tiene ningún sentido.
- El aumento de la variable *age* en una unidad hace que el precio de la vivienda incremente 0.01€
- El aumento de la variable *teachers* en una unidad hace que el precio de la vivienda incremente 0.91€
- El aumento de la variable *poor_prop* en una unidad hace que el precio de la vivienda disminuya 0.59€
- El aumento de la variable *room_num* en una unidad hace que el precio de la vivienda aumente 4.44€
- El aumento de la variable *n_hos_beds* en una unidad, hace que el precio de la vivienda disminuya 0.01€
- La existencia de *aeropuerto* o no, hace que el precio de la vivienda disminuya 0.75€

2.3.2. ¿Es significativamente mejor el nuevo modelo?

```
rlm_3 <- function(age,teachers,poor_prop,room_num,n_hos_beds,n_hot_rooms,airport){
    return (B0 + B1*age + B2*teachers + B3*poor_prop + B4*room_num + B5*n_hos_beds + B6*n_hot_rooms + B
}
predict_rlm_3 <- function(age,teachers,poor_prop,room_num,n_hos_beds,n_hot_rooms,airport){
    y <- c()
    for (i in 1:length(age)){
        y[i] <- rlm_3(age[i],teachers[i],poor_prop[i],room_num[i],n_hos_beds[i],n_hot_rooms[i],airport[
    }
    return(y)
}
y_predict_rlm3 = predict_rlm_3(house$age,house$teachers,house$poor_prop,house$room_num,
                               house$n_hos_beds,house$n_hot_rooms,house$airport)

get_r_square(y, y_predict_rlm3)
```

```
##           [,1]
## [1,] 0.6931885
```

En este caso, el coeficiente de determinación nos indica que el modelo logra explicar un 69.3% la variabilidad del precio de las viviendas, sin embargo el modelo anterior lograba explicar un 69.1%, lo que indica que la inclusión de la variable *airport*, no ha mejorado prácticamente nada el modelo. Esto se debe a que esta variable es de tipo cualitativa y la regresión que se está realizando es lineal, por lo que es difícil que una variable cualitativa ayude a mejorar a un modelo de regresión lineal.

2.3.3. Efectuar una predicción del precio de la vivienda.

Enunciado:

Para una vivienda cuyas características son: age =70, teachers =15 , poor_prop =15, room_num =8, n_hos_beds=8, n_hot_rooms=100 Utilizar el modelo Model.2.2

```
predict_rlm_2(70,15,15,8,8,100)
```

```
## [1] 23.20824
```

El precio obtenido es de 50.51, que en el caso de estar en unidades de 10 mil euros, sería 50 mill 510 euros.

2.3.4. Efectuar una verificación visual de las suposiciones de modelización.

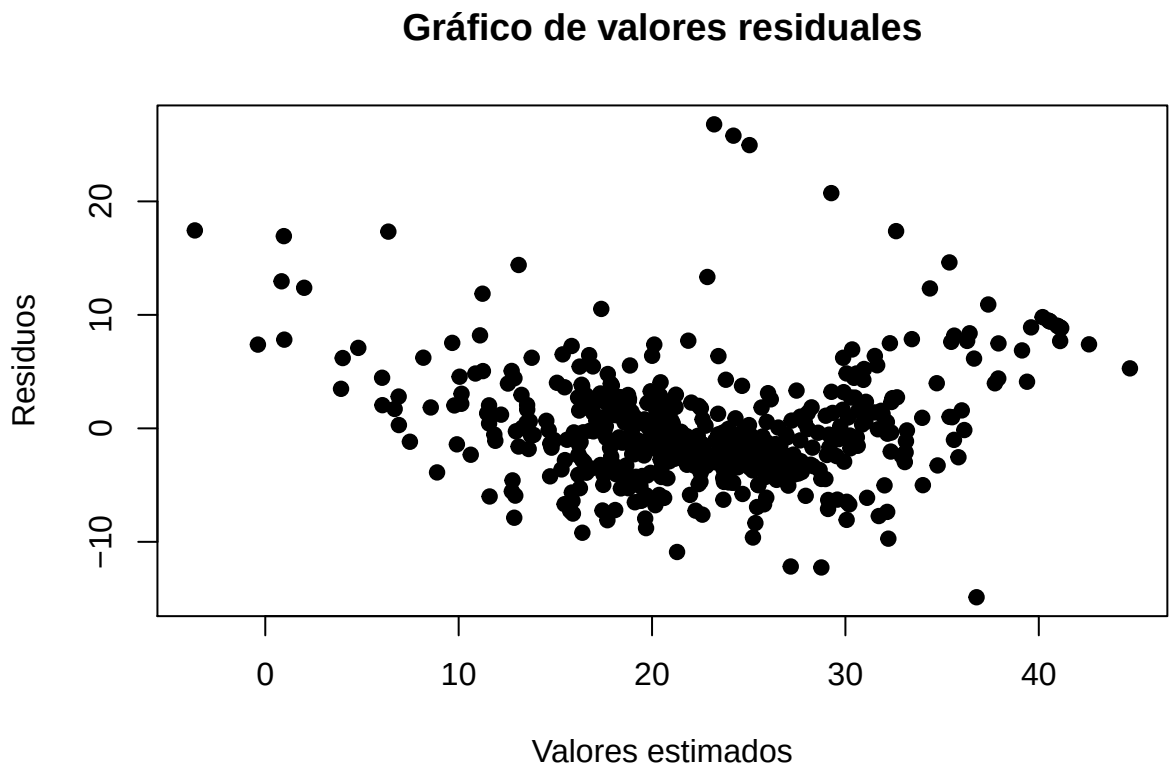
Enunciado:

Analiza los residuos del modelo. Comenta los resultados.

Solución:

A continuación, se procede a representar los residuos obtenidos por el último modelo calculado

```
residuos_rlm3 = y - y_predict_rlm3

plot(y_predict_rlm3, residuos_rlm3, main="Gráfico de valores residuales",
     xlab="Valores estimados", ylab="Residuos", pch=19)
```



Como se puede observar, los residuos se distribuyen sin mostrar ninguna forma aparentemente, por lo que indica que el modelo no está sesgado hacia ningún lado.

3. Modelo de regresión logística

Enunciado:

Se desea ajustar un modelo predictivo para predecir la expectativa que una vivienda sea vendida y conocer los factores influyentes en la predicción.

Convertir la variable Sold a tipo factor y recodificar los valores, asignando "Not" al 0 y "Yes" al 1.

Solución:

```
head(house$Sold)
```

```
## [1] 0 0 0 0 0 0
## Levels: 0 1
```

```
tail(house$Sold)
```

```
## [1] 1 1 1 1 1 1
## Levels: 0 1
```

```
levels(house$Sold)[levels(house$Sold)=="0"] <- "Not"
levels(house$Sold)[levels(house$Sold)=="1"] <- "Yes"
head(house$Sold)
```

```
## [1] Not Not Not Not Not Not
## Levels: Not Yes
```

```r
tail(house$Sold)
```

```
## [1] Yes Yes Yes Yes Yes Yes
## Levels: Not Yes
```

3.1. Regresores cuantitativos

3.1.1. Calcular

Enunciado:

Estimar el modelo de regresión logística donde la variable dependiente es Sold y las explicativas price, age, poor_prop

Solución: Se procede a continuación a calcular el modelo de regresión logística a través de la función glm:

```r
glm_1 <- glm(formula = house$Sold ~ house$price + house$age + house$poor_prop,
             family = binomial (link=logit))
summary(glm_1)
```

```
##
## Call:
## glm(formula = house$Sold ~ house$price + house$age + house$poor_prop,
##     family = binomial(link = logit))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7699  -1.0968  -0.4659   1.1130   1.8768
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      4.565208   0.711571   6.416 1.40e-10 ***
## house$price     -0.139051   0.020462  -6.796 1.08e-11 ***
## house$age        0.009405   0.004527   2.077   0.0378 *
## house$poor_prop -0.184734   0.029050  -6.359 2.03e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 686.12  on 497  degrees of freedom
## Residual deviance: 618.86  on 494  degrees of freedom
## AIC: 626.86
##
## Number of Fisher Scoring iterations: 4
```

3.1.2. Interpretar

Enunciado:

Estima los odds ratio de las variables price, age, poor_prop mediante un intervalo de confianza del 95 % e interpreta los intervalos obtenidos. ¿Cuál sería el odds ratio de un quinquenio?

Solución:

```r
exp(confint(glm_1))
```

```
## Waiting for profiling to be done...
```

```
##                       2.5 %      97.5 %
## (Intercept)     24.9829433 408.6853792
## house$price      0.8343942   0.9042313
## house$age        1.0005767   1.0185206
## house$poor_prop  0.7835919   0.8782422
```

La interpretación de los resultados obtenidos es la siguiente:

- por cada unidad que aumente el **precio** de la vivienda, el odds de vender la vivienda es entre 0,83 y 0,9 veces menor
- por cada unidad que aumente la **edad** de la vivienda, el odds de vender la vivienda es entre 1.00 y 1.01 veces mayor
- por cada unidad que aumente la **proporción de población pobre** en la ubicación donde se encuentra la vivienda, el odds de vender la vivienda es entre 0.78 y 0.87 veces mayor

```
exp(coefficients(glm_1))
```

```
##     (Intercept)     house$price       house$age house$poor_prop
##      96.0826011       0.8701834       1.0094494       0.8313250
```

Según el resultado de la celda anterior, si la edad aumenta en 5 unidades, el odds será $1.009^5 = 1.045$ veces mayor.

3.2. Regresores cualitativos

3.2.1. Calcular

Enunciado:

Estimar el modelo de regresión logística donde la variable dependiente es Sold y la explicativa airport

Solución: Se procede a continuación a calcular el modelo de regresión logística a través de la función glm:

```
head(house$airport)
```

```
## [1] 0 0 0 0 0 0
```

```
glm_2 <-glm(formula = house$Sold ~ house$airport, family = binomial (link=logit))
summary(glm_2)
```

```
##
## Call:
## glm(formula = house$Sold ~ house$airport, family = binomial(link = logit))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2440  -1.1661  -0.9322   1.1122   1.4443
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -0.6084     0.1389  -4.379 1.19e-05 ***
## house$airport   0.7637     0.1848   4.133 3.59e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 686.12  on 497  degrees of freedom
## Residual deviance: 668.67  on 496  degrees of freedom
```

```
## AIC: 672.67
##
## Number of Fisher Scoring iterations: 4
```

3.2.2. Interpretar

Enunciado:

Estima el odds ratio de la variable airport mediante un intervalo de confianza del 95 % e interpreta el intervalo obtenido.

Solución:

```
exp(confint(glm_2))
```

```
## Waiting for profiling to be done...
```

```
##                    2.5 %    97.5 %
## (Intercept)   0.4127939 0.7122509
## house$airport 1.4972103 3.0912440
```

La interpretación de los resultados obtenidos es la siguiente: * la venta de una vivienda en un lugar donde haya un **aeropuerto** es entre 1.49 y 3.09 veces más probable que en un lugar donde no lo haya

3.3. Regresores cuantitativos y cualitativos

Enunciado:

Estimar el modelo de regresión logística donde la variable dependiente es Sold y los regresores price, age, poor_prop y airport.

Solución: Se procede a continuación a calcular el modelo de regresión logística a través de la función glm:

```
glm_3 <- glm(formula = Sold ~ price + age + poor_prop + airport, data=house,
             family = binomial (link=logit))
summary(glm_3)
```

```
##
## Call:
## glm(formula = Sold ~ price + age + poor_prop + airport, family = binomial(link = logit),
##     data = house)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8248  -1.0648  -0.3795   1.0850   1.8975
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.315327   0.748326   5.767 8.09e-09 ***
## price       -0.148835   0.022089  -6.738 1.60e-11 ***
## age          0.009618   0.004614   2.084   0.0371 *
## poor_prop   -0.186421   0.029884  -6.238 4.43e-10 ***
## airport      0.830196   0.201939   4.111 3.94e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 686.12  on 497  degrees of freedom
```

```
## Residual deviance: 601.43  on 493  degrees of freedom
## AIC: 611.43
##
## Number of Fisher Scoring iterations: 4
```

3.3.1. Interpretar

Enunciado:

Estima los odds ratio de las variables regresoras mediante un intervalo de confianza del 95 % e interpreta los intervalos obtenidos. ¿Qué regresor tiene más impacto en la probabilidad de venta?

Solución:

```
exp(confint(glm_3))
```

```
## Waiting for profiling to be done...
```

```
##                   2.5 %      97.5 %
## (Intercept) 18.0950956 341.9562881
## price        0.8234356   0.8980463
## age          1.0006246   1.0189159
## poor_prop    0.7808973   0.8780886
## airport      1.5491740   3.4219779
```

La interpretación de los resultados obtenidos es la siguiente:

- por cada unidad que aumente el **precio** de la vivienda, el odds de vender la vivienda es entre 0,82 y 0,89 veces menor
- por cada unidad que aumente la **edad** de la vivienda, el odds de vender la vivienda es entre 1.00 y 1.01 veces mayor
- por cada unidad que aumente la **proporción de población pobre** en la ubicación donde se encuentra la vivienda, el odds de vender la vivienda es entre 0.78 y 0.87 veces mayor
- la venta de una vivienda en un lugar donde haya un **aeropuerto** es entre 1.54 y 3.42 veces más probable que en un lugar donde no lo haya

Con respecto a la pregunta *¿Qué regresor tiene más impacto en la probabilidad de venta?*, el regresor que presenta más impacto en la probabilidad de venta es el correspondiente con la variable *airport*, lo que quiere decir que **lo que más impacto tiene en la probabilidad de venta de una casa, es la existencia de un aeropuerto cercano a la misma**.

3.3.2. Predicción de venta

Enunciado:

Para una vivienda cuyas características son: price=20, age=50, poor_prop=50 y airport= YES.

Solución: Se procede a continuación a realizar la predicción solicitada en el enunciado a través de la función predict:

```
predict(glm_3, newdata = data.frame(price=20,age=50,poor_prop=50,airport=1),type='response')
```

```
##           1
## 0.001265094
```

Como se puede observar, la probabilidad de venta obtenida para una vivienda cuyas características son: *price=20, age=50, poor_prop=50 y airport= YES.* es de un 0.0012, lo que indica que la probabilidad de venta es muy baja.

3.3.3. Estimación por resustitución de la precisión del modelo

Enunciado:

Proporcionar la tabla de confusión correspondiente al modelo. Comenta los resultados.

Solución:

Se procede a continuación a calcular la tabla de confusión a través de la función ConfusionMatrix:

```
y_pred <- ifelse(glm_3$fitted.values < 0.5, 0, 1)
glm_3.confusion_matrix = ConfusionMatrix(y_true = house$Sold,y_pred = y_pred)
glm_3.confusion_matrix
```

```
##       y_pred
## y_true   0   1
##    Not 181  91
##    Yes 100 126
```

Como se puede observar, existen 91 casos de falsos negativos y 100 de falsos positivos frente a 181 casos verdaderos negativos y 126 casos verdaderos positivos. Esto indica que del total de la muestra a predecir, se están prediciendo correctamente 307 casos y se están prediciendo de manera errónea los 191 casos restantes.

Además, los 91 casos de falsos negativos indican que 91 casos han sido predichos como que no se va a vender la vivienda y realmente si se vendió y los y los 100 falsos positivos indican que 100 casos han sido predichos como que se va a vender la vivienda y realmente no se vendió.

3.3.4. Visualización

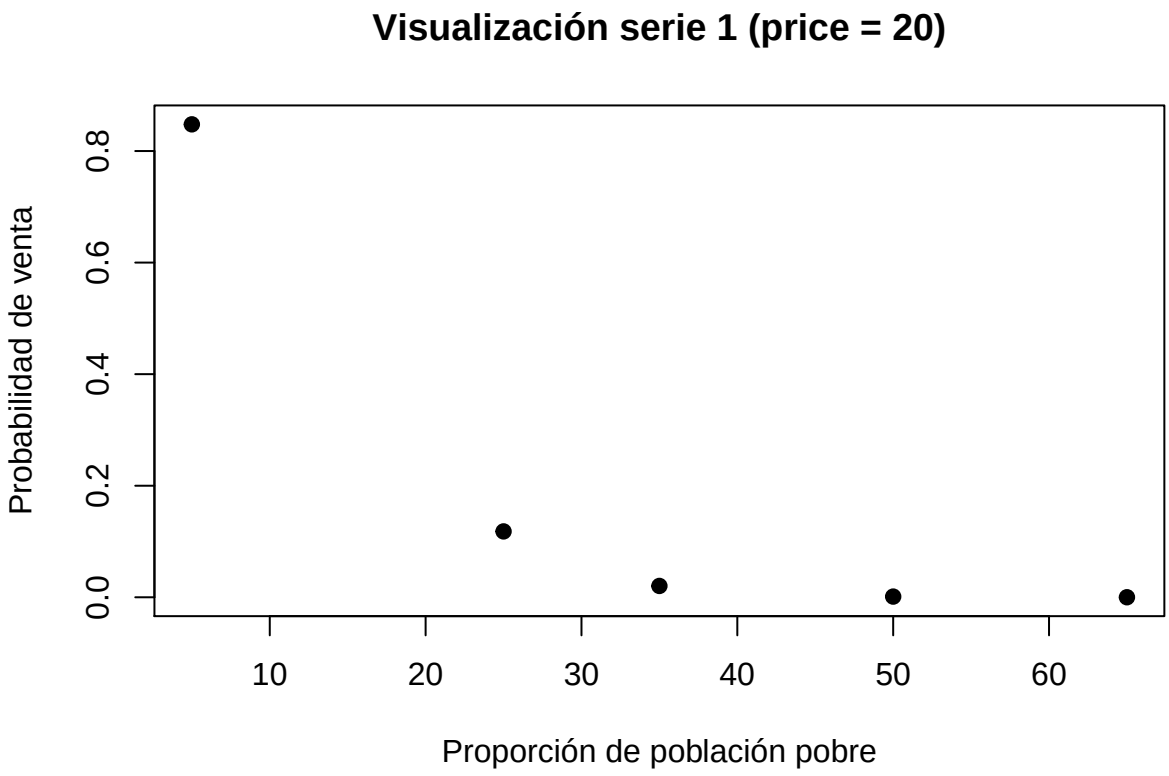
Enunciado:

Para los distintos valores de la variable price = c(20,30,40) se representaran las tres series de probabilidades de venta en un mismo gráfico de dispersión XY. En concreto, para cada valor de price, se tomarán los valores fijos de age =50, airport = "YES", y se representarán las probabilidades de venta (eje Y) para los valores de poor_prop = c(5,25,35,50,65) (eje X). Comenta el gráfico obtenido.

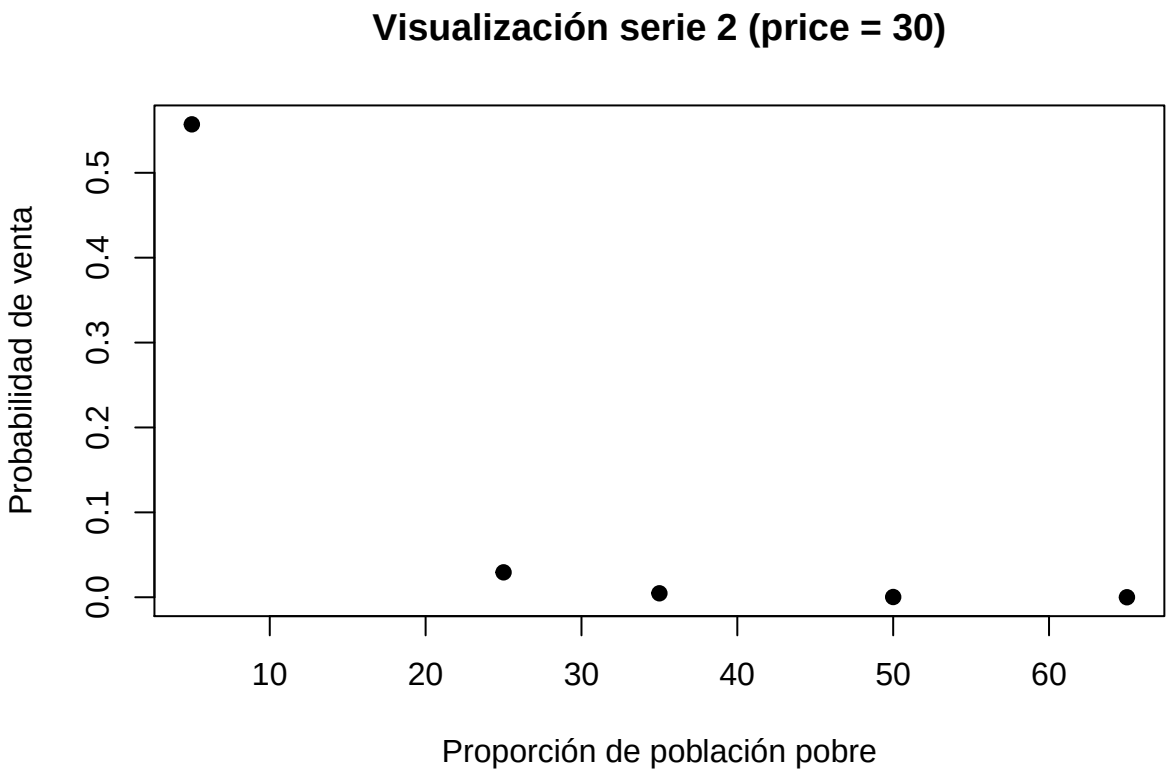
Solución

Se procede a continuación a dibujar el gráfico solicitado en el enunciado.

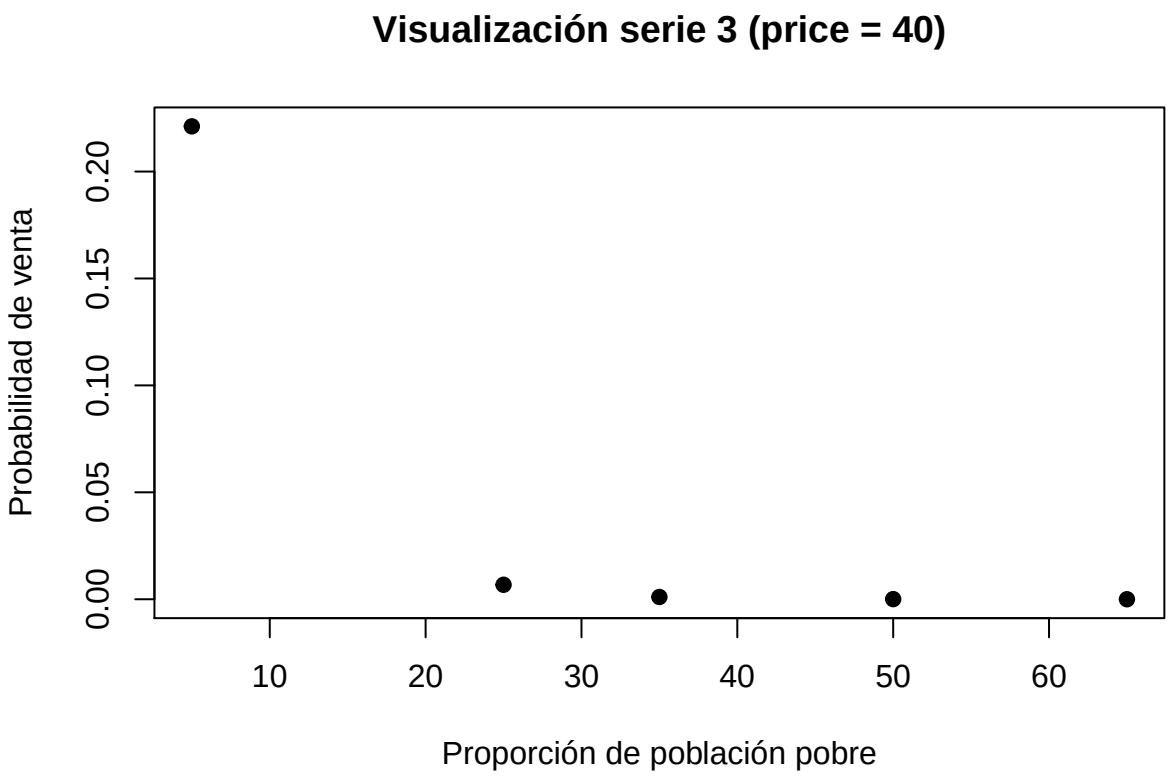
```
df = data.frame(price=0,age=0,airport=0,poor_prop=0)
price <- c(20,30,40)
age <- 50
airport <- 1
poor_prop <- c(5,25,35,50,65)

for (i in 1:length(price)){
    for (j in 1:length(poor_prop)){
        df[j+(i-1)*5,]=c(price[i],age,airport,poor_prop[j])
    }
}
y_pred_dv <- predict(glm_3,newdata = df,type = 'response')
df['y_pred'] <- y_pred_dv
df_1 = df[1:5,]
df_2 = df[6:10,]
df_3 = df[11:15,]

plot(df_1$poor_prop, df_1$y_pred, main="Visualización serie 1 (price = 20)",
     ylab="Probabilidad de venta", xlab="Proporción de población pobre", pch=19)
```




```
plot(df_2$poor_prop, df_2$y_pred, main="Visualización serie 2 (price = 30)",
     ylab="Probabilidad de venta", xlab="Proporción de población pobre", pch=19)
```




```
plot(df_3$poor_prop, df_3$y_pred, main="Visualización serie 3 (price = 40)",
     ylab="Probabilidad de venta", xlab="Proporción de población pobre", pch=19)
```



Los gráficos obtenidos, nos indican que a medida que aumenta la proporción de población pobre, disminuye la probabilidad de venta de la vivienda, pero que a medida que va subiendo el precio, dicha probabilidad disminuye de manera más fuerte a medida que hay mayor porporción de población pobre. Esto indica la existencia de una gran diferencia de clases sociales, pues a las casas que cuestan más dinero, solo pueden acceder aquellas personas con una clase social más alta, y según los gráficos anteriores, a la mínima que existe una parte de la población pobre al rededor de la casa, la probabilidad de venta disminuye, lo que indica la gente de clase alta, normalmente, le gusta estar rodeada de gente de clase alta y no de la gente de clase baja.

4. Conclusión

Tras realizar esta práctica y obtener los resultados que se han obtenido de los datos, se obtiene como conclusión que tanto en el precio de la vivienda como en la probabilidad de venta de la misma, una de las variables que presenta una mayor influencia es la proporción de la población pobre. Si que en el caso de la regresión logística, influye más la existencia de un aeropuerto cercano a la vivienda, pero eso es debido a que la variable *airport* es de tipo cualitativa y la variable *poor_prop* es de tipo cuantitativa, por lo que resulta dificil que esta afecte más que una variable cualitativa.

Por otro lado, la variable *airport*, ha sido la que más influencia ha tenido sobre la probabilidad de venta, obteniendo un intervalo de 1.54 y 3.42 con un 95% de confianza. Sin embargo en la variabilidad del precio no ha tenido mucha importancia, pero porque se trata de una variable categórica, como se ha indicado anteriormente, quizá si se hubiese tratado de buscar alguna relación diferente a una lineal, si que hubiera tenido influencia.