Rental Price Prediction in Barcelona

Rebeca Suárez Ojeda

2025-05-12

1. Regresión lineal

1.1 Estudio de correlación lineal

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(corrplot)
```

```
## corrplot 0.95 loaded
```

```
datos <- read_csv("Barcelona_Rent_Price_vf.csv", locale = locale(encoding = "UTF-8"))
```

```
## Rows: 7949 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): real_state, neighborhood
## dbl (4): price, rooms, bathroom, square_meters
## lgl (2): lift, terrace
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
head(datos)
```

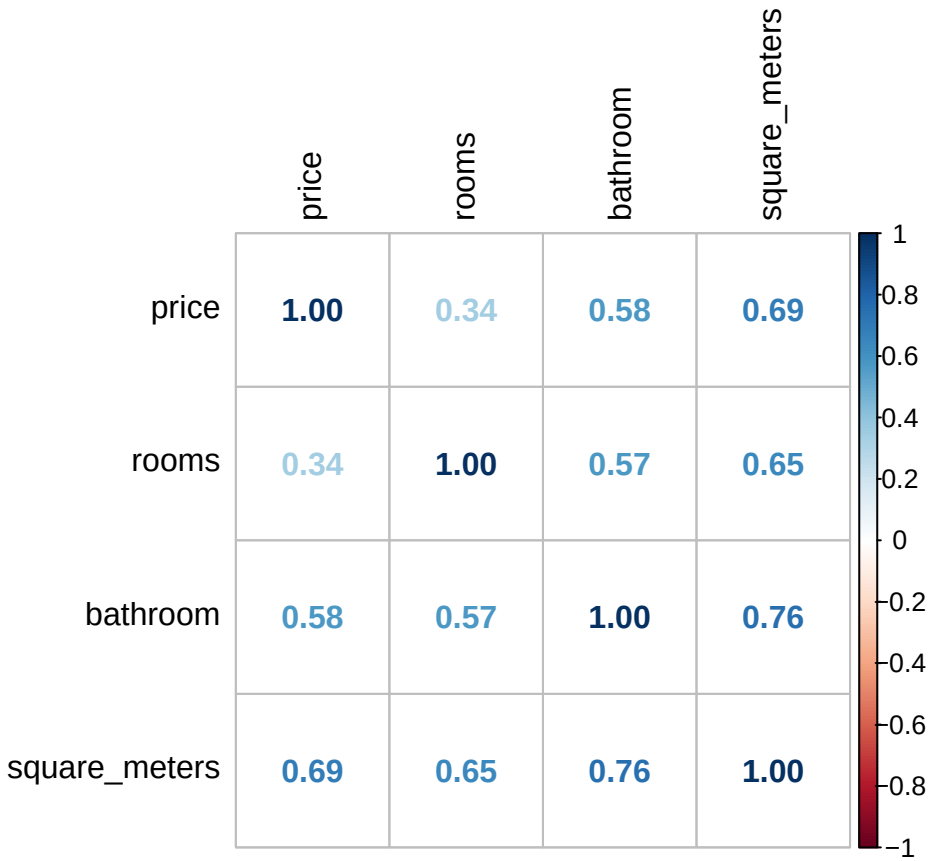
```
## # A tibble: 6 x 8
##   price rooms bathroom lift  terrace square_meters real_state neighborhood
```

```
##   <dbl> <dbl>    <dbl> <lgl> <lgl>           <dbl> <chr>      <chr>
## 1  3000     3        2 FALSE FALSE             150 apartment  Eixample
## 2  1250     1        1 FALSE FALSE              35 apartment  Eixample
## 3  2200     4        2 FALSE FALSE              90 apartment  Gracia
## 4  3468     3        2 FALSE FALSE              98 apartment  Ciutat Vella
## 5  1100     1        1 TRUE  FALSE              45 apartment  Les Corts
## 6  2500     3        2 TRUE  FALSE             180 apartment  Eixample
```

```
datos_cuantitativas <- datos %>%
  select(price, rooms, bathroom, square_meters) %>%
  drop_na()

matriz_cor <- cor(datos_cuantitativas, method = "pearson")
matriz_cor
```

```
##                    price     rooms  bathroom square_meters
## price          1.0000000 0.3433956 0.5788707     0.6927869
## rooms          0.3433956 1.0000000 0.5741637     0.6467779
## bathroom       0.5788707 0.5741637 1.0000000     0.7559238
## square_meters 0.6927869 0.6467779 0.7559238     1.0000000
```

```
corrplot(matriz_cor, method = "number", tl.col = "black")
```



```r
cor_precio <- sort(matriz_cor["price", -1], decreasing = TRUE)
cor_precio
```

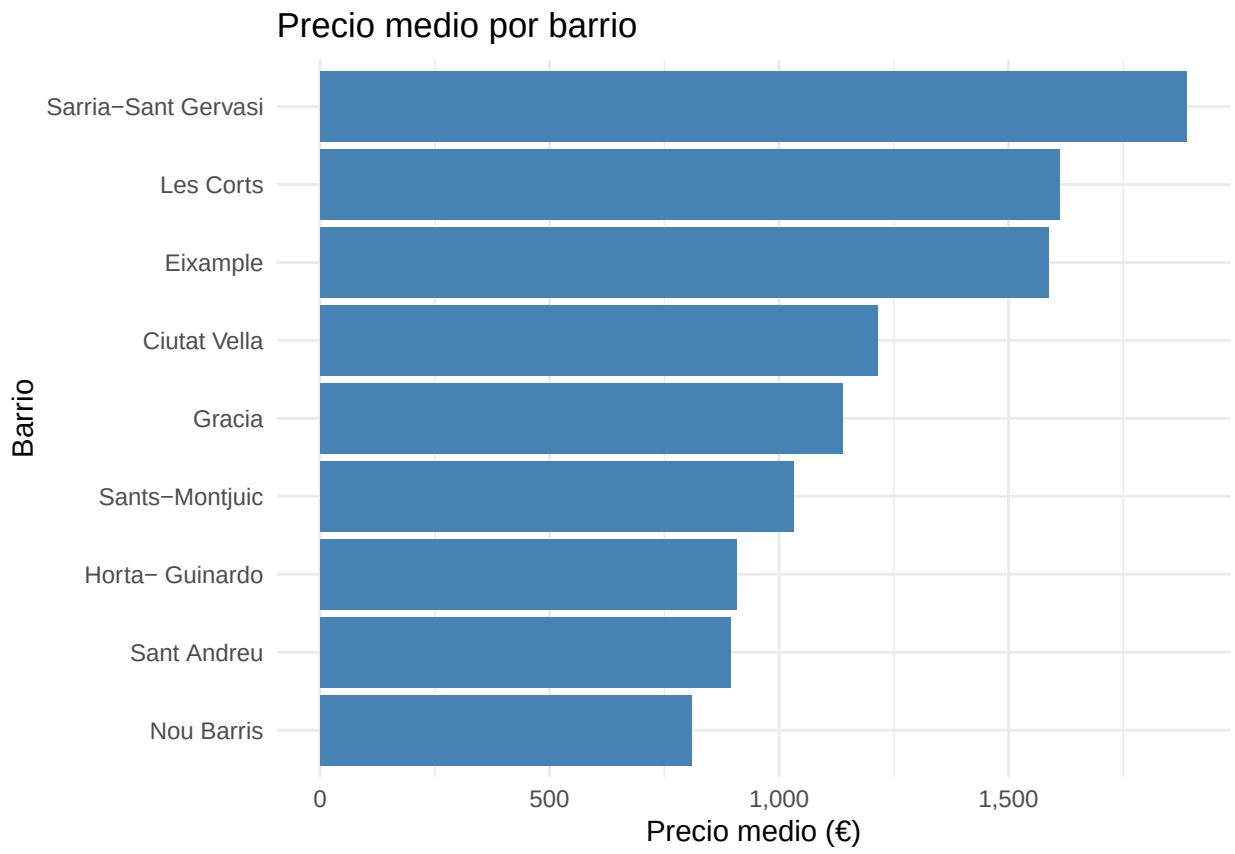
```
## square_meters      bathroom         rooms
##     0.6927869     0.5788707     0.3433956
```

- square_meters es la variable que tiene mayor correlacion lineal positiva con el precio del alquiler; cuanto mayor es la superficie, mayor es el precio
- bathroom también presenta una correlacion moderada, cuanto mas baños mayor precio
- rooms tiene una correlacion mas débil con el precio, posiblemente porque hay viviendas con muchas habitaciones pero pequeñas o mal distribuidas.

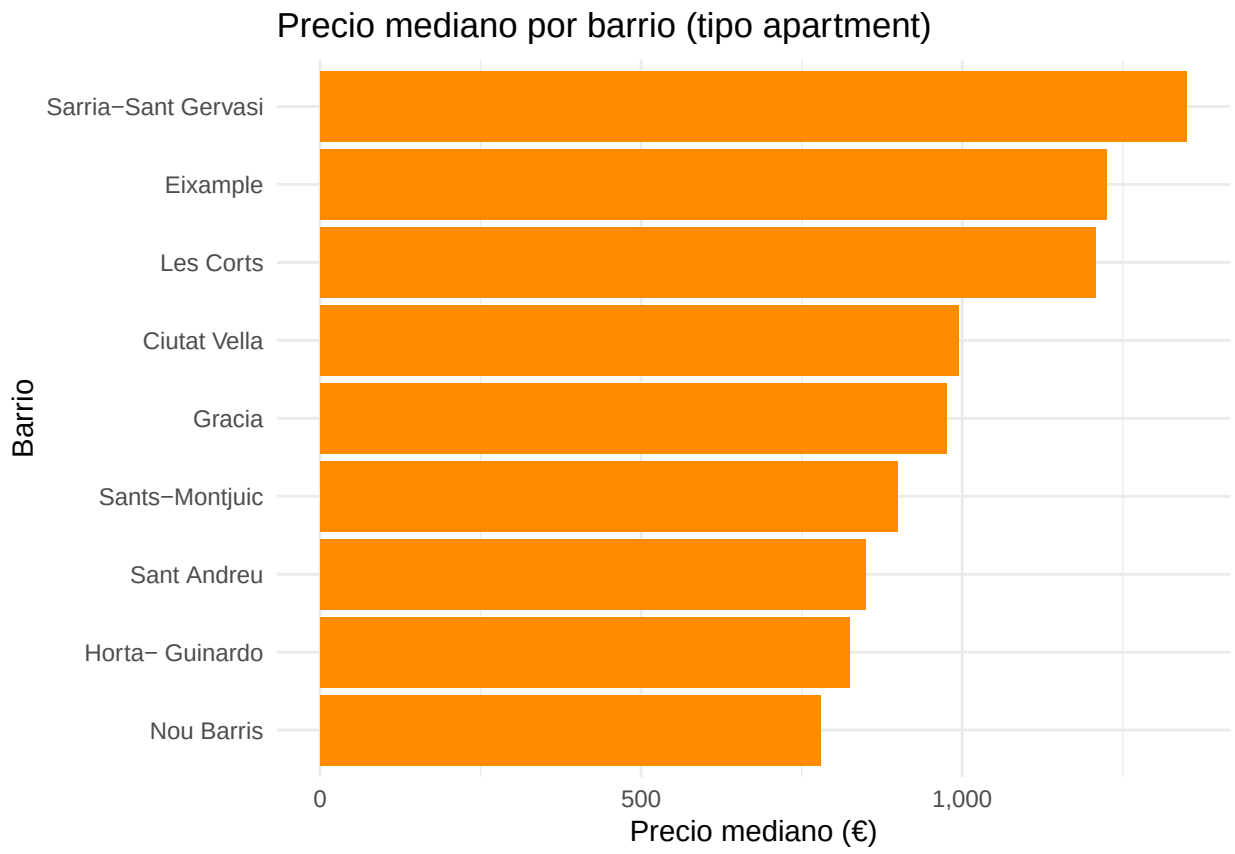
Esto indica que el tamaño de la vivienda es el factor cuantitativo más importante para predecir el precio del alquiler.

1.2 Estudio comparativo entre barrios de Barcelona

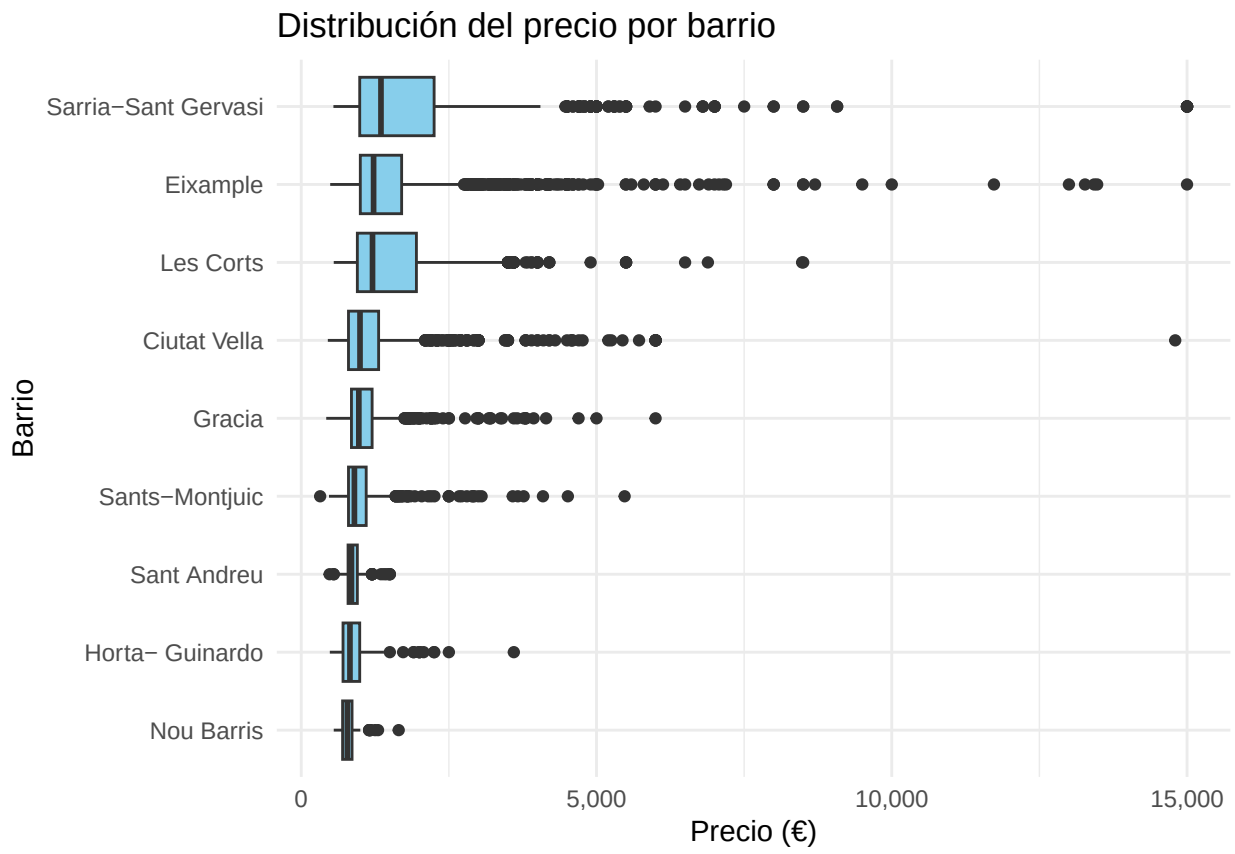
```r
datos <- datos %>%
  mutate(neighborhood = iconv(neighborhood, from = "", to = "UTF-8"))

datos_filtrados <- datos %>%
  filter(!is.na(price), !is.na(neighborhood), neighborhood != "")

datos_resumen <- datos_filtrados %>%
  group_by(neighborhood) %>%
  summarise(precio_medio = mean(price, na.rm = TRUE))

ggplot(datos_resumen, aes(x = reorder(neighborhood, precio_medio), y = precio_medio)) +
  geom_col(fill = "steelblue") +
  coord_flip() +
  labs(title = "Precio medio por barrio",
       x = "Barrio", y = "Precio medio (€)") +
  scale_y_continuous(labels = scales::comma) +
  theme_minimal()
```




```r
datos_mediana <- datos_filtrados %>%
  group_by(neighborhood) %>%
  summarise(precio_mediano = median(price, na.rm = TRUE))

ggplot(datos_mediana, aes(x = reorder(neighborhood, precio_mediano), y = precio_mediano)) +
  geom_col(fill = "darkorange") +
  coord_flip() +
  labs(
    title = "Precio mediano por barrio (tipo apartment)",
    x = "Barrio",
    y = "Precio mediano (€)"
  ) +
  scale_y_continuous(labels = scales::comma) +
  theme_minimal()
```




```r
ggplot(datos_filtrados, aes(x = reorder(neighborhood, price, FUN = median), y = price)) +
  geom_boxplot(fill = "skyblue") +
  coord_flip() +
  labs(
    title = "Distribución del precio por barrio",
    x = "Barrio",
    y = "Precio (€)"
  ) +
  scale_y_continuous(labels = scales::comma) +
  theme_minimal()
```



- Sarria-Sant Gervasi lidera consistentemente como el barrio más caro, tanto en media como en mediana
- Les Corts y Eixample le siguen muy de cerca, aunque Eixample tiene más dispersión.
- Nou Barris es el barrio más asequible, con poca variación de precios.
- Hay una diferencia clara de precio según el barrio, lo que sugiere que el factor "neighborhood" será clave en los modelos predictivos posteriores.

1.3 Generación de los conjuntos de entrenamiento y de test

```
set.seed(123)

datos_modelo <- datos %>%
  filter(!is.na(price), !is.na(rooms), !is.na(bathroom), !is.na(square_meters),
         !is.na(lift), !is.na(terrace), !is.na(real_state), !is.na(neighborhood))

n <- nrow(datos_modelo)
train_indices <- sample(seq_len(n), size = 0.8 * n)

datos_train <- datos_modelo[train_indices, ]
datos_test <- datos_modelo[-train_indices, ]

cat("Tamaño conjunto de entrenamiento:", nrow(datos_train), "\n")
```

```
## Tamaño conjunto de entrenamiento: 5883
```

```
cat("Tamaño conjunto de prueba:", nrow(datos_test), "\n")
```

```
## Tamaño conjunto de prueba: 1471
```

1.4 Estimación del modelo de regresión lineal con predictores cuantitativos

```
modelo_cuant <- lm(price ~ rooms + bathroom + square_meters, data = datos_train)
summary(modelo_cuant)
```

```
##
## Call:
## lm(formula = price ~ rooms + bathroom + square_meters, data = datos_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3184.2  -300.0  -113.4   109.3 11562.9
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    170.2162    24.0250   7.085 1.55e-12 ***
## rooms         -172.3479    11.1840 -15.410  < 2e-16 ***
## bathroom       307.1931    19.8967  15.439  < 2e-16 ***
## square_meters   14.2303     0.3407  41.765  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 721.7 on 5879 degrees of freedom
## Multiple R-squared:  0.5136, Adjusted R-squared:  0.5134
## F-statistic:  2070 on 3 and 5879 DF,  p-value: < 2.2e-16
```

El modelo resultante muestra que:

- La variable square_meters tiene una fuerte influencia positiva sobre el precio (coef = 14.23).
- bathroom también incrementa el precio (coef = 307.19).
- rooms muestra un coeficiente negativo (coef = –172.35), lo que podría deberse a colinealidad con otras variables.

El modelo explica aproximadamente el 51% de la variabilidad del precio (R^2 ajustado = 0.5134), y todos los coeficientes son estadísticamente significativos ($p < 0.001$).

1.4.1 Comprobación de colinealidad

```
library(car)
```

```
## Cargando paquete requerido: carData
```

```
##
## Adjuntando el paquete: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode

## The following object is masked from 'package:purrr':
##
##     some
```

```r
vif(modelo_cuant)
```

```
##         rooms      bathroom square_meters
##      1.859046      2.453813      2.950425
```

Se ha comprobado la colinealidad entre las variables rooms, bathroom y square_meters mediante el cálculo del VIF. Todos los factores tienen valores por debajo de 4 (rooms = 1.86, bathroom = 2.45, square_meters = 2.95), por lo que no se detecta colinealidad significativa.

En consecuencia, no se elimina ninguna variable del modelo.

1.5 Estimación del modelo de regresión lineal con predictores cuantitativos y cualitativos

```r
library(dplyr)

datos_train$lift <- as.factor(datos_train$lift)
datos_train$terrace <- as.factor(datos_train$terrace)
datos_train$real_state <- as.factor(datos_train$real_state)
datos_train$neighborhood <- relevel(as.factor(datos_train$neighborhood), ref = "Sarria-Sant Gervasi")

modelo_completo <- lm(price ~ rooms + bathroom + square_meters +
                        lift + terrace + real_state + neighborhood,
                      data = datos_train)


summary(modelo_completo)
```

```
##
## Call:
## lm(formula = price ~ rooms + bathroom + square_meters + lift +
##     terrace + real_state + neighborhood, data = datos_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3116.0  -272.2   -43.3   151.3 10669.1
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)                  895.0550    39.9587  22.399  < 2e-16 ***
## rooms                       -123.1703    10.8868 -11.314  < 2e-16 ***
## bathroom                     275.9541    18.5276  14.894  < 2e-16 ***
## square_meters                 13.8052     0.3233  42.706  < 2e-16 ***
## liftTRUE                    -140.4818    20.9811  -6.696 2.35e-11 ***
## terraceTRUE                   81.1206    23.0422   3.521 0.000434 ***
## real_stateattic             -420.0618    51.5784  -8.144 4.63e-16 ***
## real_stateflat              -727.8111    27.4895 -26.476  < 2e-16 ***
## real_statestudy             -776.0049    77.7241  -9.984  < 2e-16 ***
## neighborhoodCiutat Vella    -143.3963    31.0925  -4.612 4.07e-06 ***
## neighborhoodEixample          45.0317    26.1398   1.723 0.084991 .
## neighborhoodGracia          -111.0800    36.7595  -3.022 0.002524 **
## neighborhoodHorta- Guinardo -261.9584    48.5482  -5.396 7.09e-08 ***
## neighborhoodLes Corts       -116.9268    38.4849  -3.038 0.002390 **
## neighborhoodNou Barris      -233.7221    70.8235  -3.300 0.000972 ***
## neighborhoodSant Andreu     -223.3691    59.7734  -3.737 0.000188 ***
## neighborhoodSants-Montjuic  -126.6336    38.7325  -3.269 0.001084 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 664.2 on 5866 degrees of freedom
## Multiple R-squared:  0.5889, Adjusted R-squared:  0.5878
## F-statistic: 525.2 on 16 and 5866 DF,  p-value: < 2.2e-16
```

```
vif(modelo_completo)
```

```
##                   GVIF Df GVIF^(1/(2*Df))
## rooms         2.079394  1        1.442011
## bathroom      2.511652  1        1.584819
## square_meters 3.134896  1        1.770564
## lift          1.251089  1        1.118521
## terrace       1.143108  1        1.069162
## real_state    1.281897  3        1.042259
## neighborhood  1.361085  8        1.019454
```

Se ha ajustado un modelo de regresión lineal múltiple incluyendo variables cuantitativas y cualitativas. El R2 ajustado ha mejorado de 0.5134 a 0.5878 lo que indica que el nuevo modelo explica casi un 59% de la variabilidad del precio, una mejora respecto al modelo solo con las variables numéricas.

Según los p-valores, casi todas las varaibles son estadísticamente significativas, a exepción neighborhoodEixample que tiene un valor mayor a 0.05.

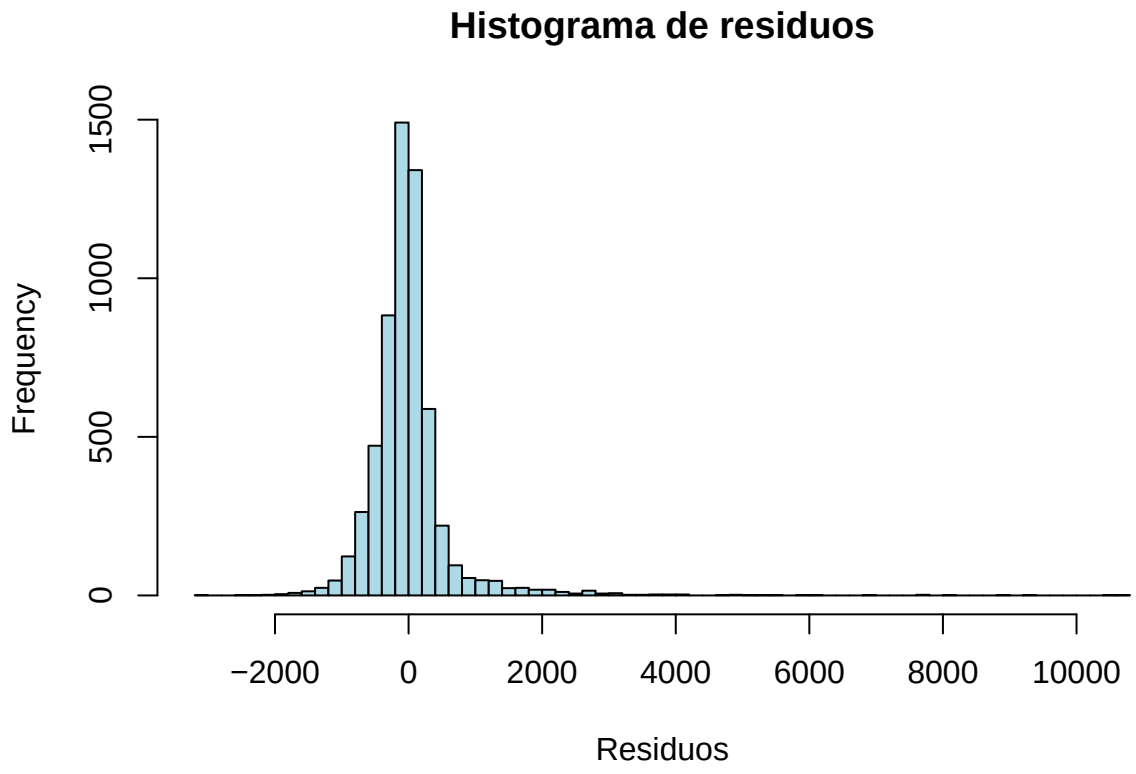
Además, la comprobación del VIF confirma quer no existe multicolinealidad grave entre las variables.

Por tanto, mantenemos todas las variables a excepcion de neighborhoodEixample que no es significativa.

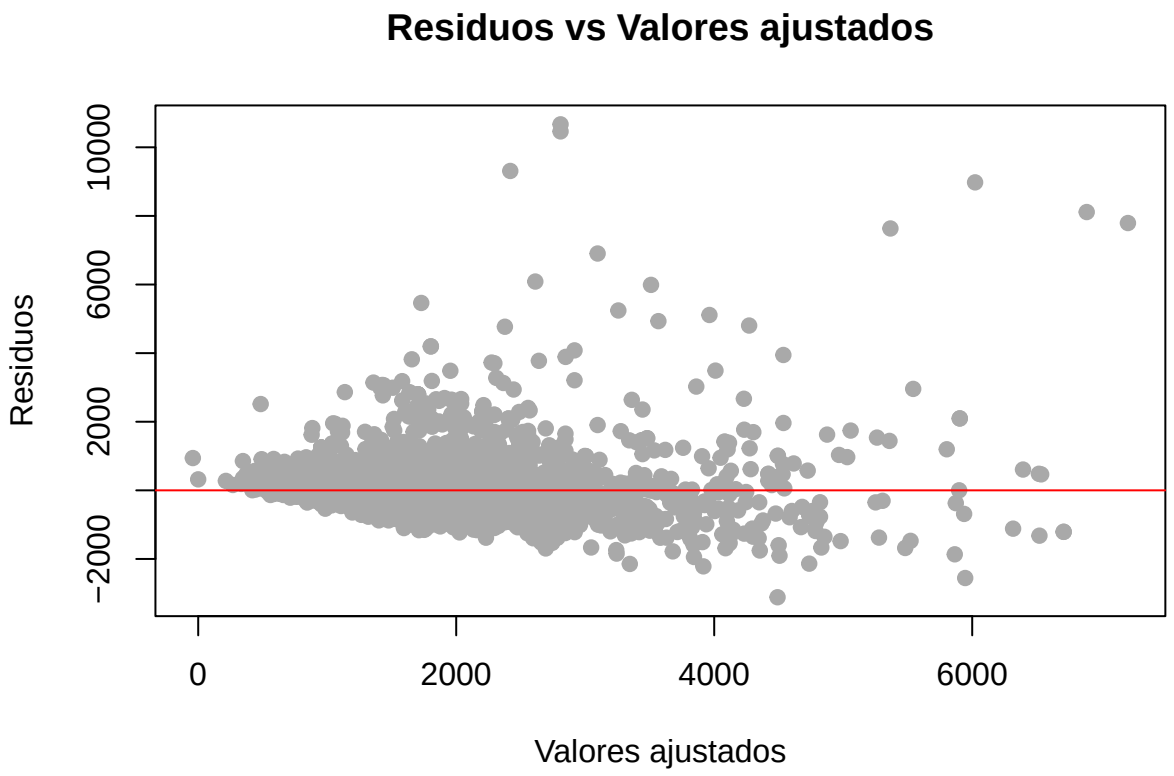
1.6 Diagnosis del modelo.

```
residuos <- resid(modelo_completo)
valores_ajustados <- fitted(modelo_completo)

hist(residuos,
```

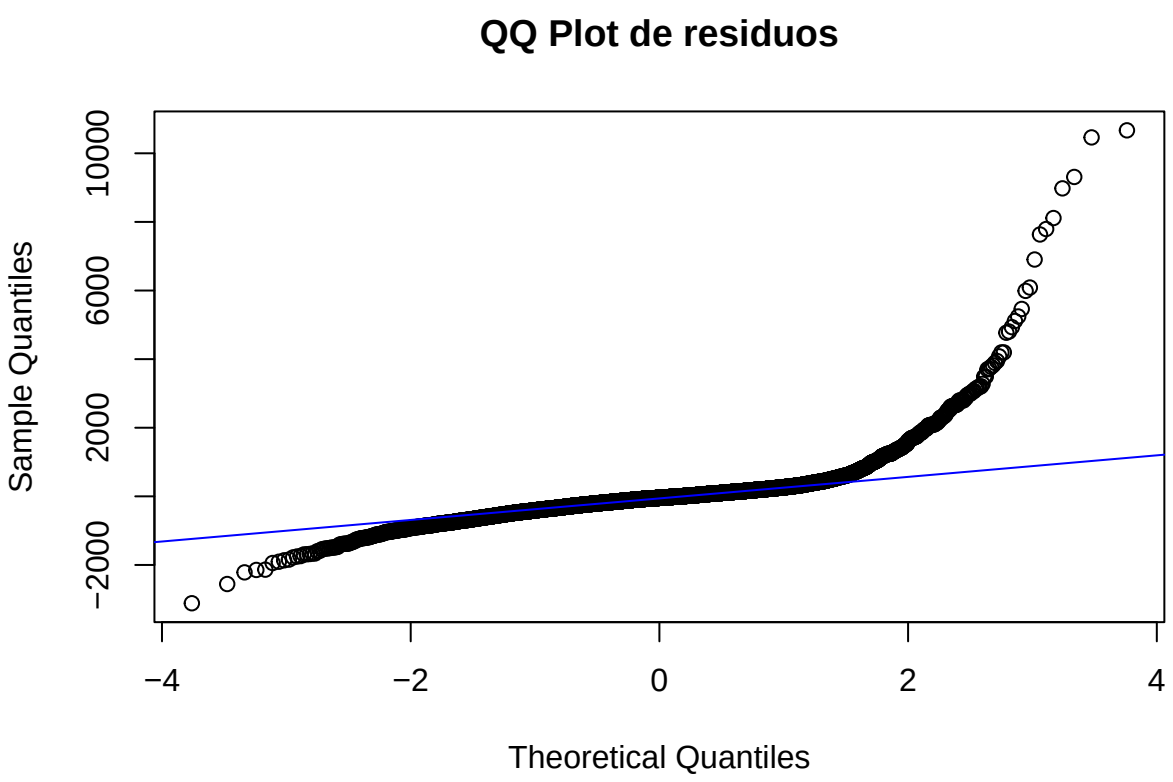
```
     breaks = 50,
     col = "lightblue",
     main = "Histograma de residuos",
     xlab = "Residuos")
```




```
plot(valores_ajustados, residuos,
     main = "Residuos vs Valores ajustados",
     xlab = "Valores ajustados",
     ylab = "Residuos",
     pch = 19, col = "darkgrey")
abline(h = 0, col = "red")
```




```r
qqnorm(residuos, main = "QQ Plot de residuos")
qqline(residuos, col = "blue")
```



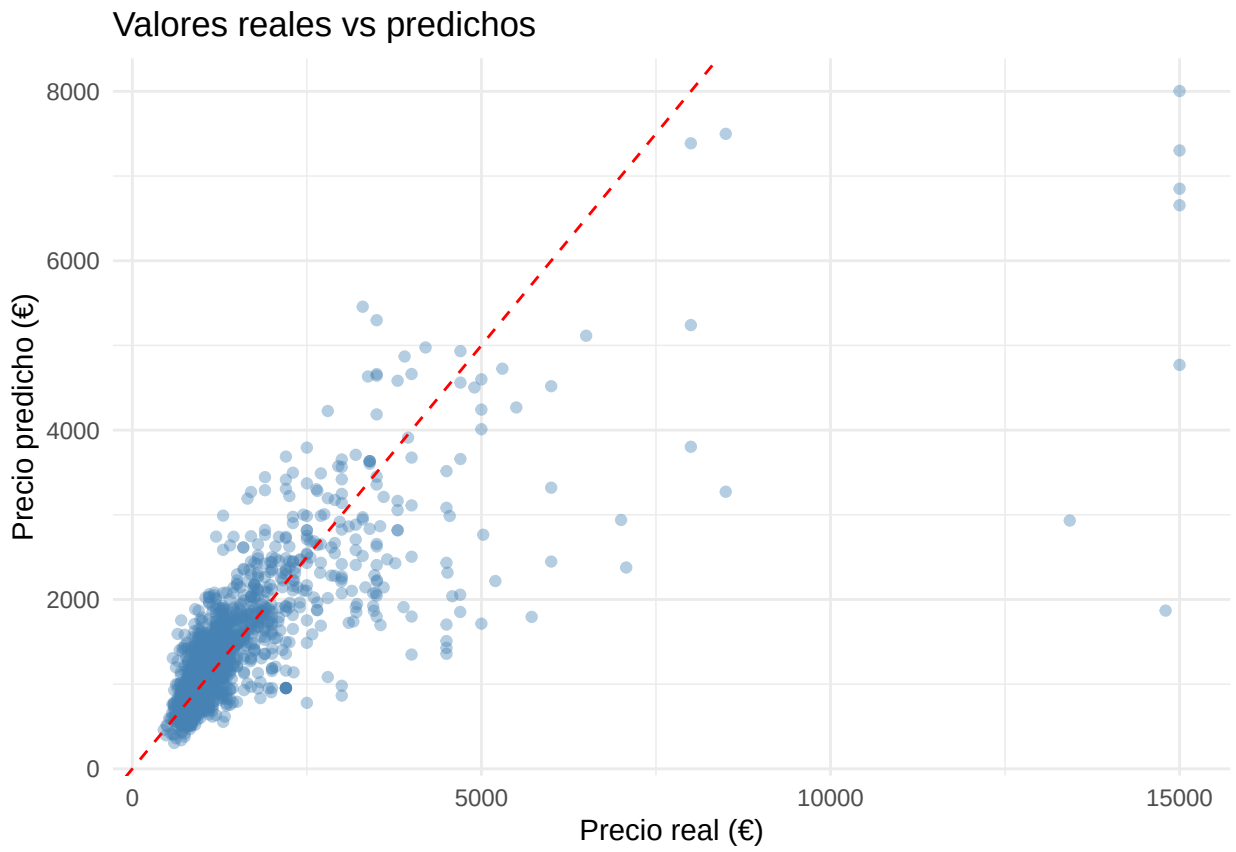
- El histograma muestra una distribución aproximadamente normal, con una ligera asimetría hacia la derecha
- El gráfico de residuos frente a los valores ajustados sugiere que los errores tienden a aumentar con el valor del alquiler, indicando cierta heterocedasticidad.
- El QQ plot confirma que los residuos se ajustan en su mayoría a una distribución normal, aunque hay outliers que generan colas más pesadas.

1.7 Predicción del modelo.

```
datos_test$lift <- as.factor(datos_test$lift)
datos_test$terrace <- as.factor(datos_test$terrace)
datos_test$real_state <- as.factor(datos_test$real_state)
datos_test$neighborhood <- relevel(as.factor(datos_test$neighborhood), ref = "Sarria-Sant Gervasi")

predicciones <- predict(modelo_completo, newdata = datos_test)

reales <- datos_test$price

rmse <- sqrt(mean((predicciones - reales)^2))
cat("RMSE:", round(rmse, 2))
```

```
## RMSE: 882.61
```

```r
library(ggplot2)

df_resultados <- data.frame(Reales = reales, Predichos = predicciones)

ggplot(df_resultados, aes(x = Reales, y = Predichos)) +
  geom_point(alpha = 0.4, color = "steelblue") +
  geom_abline(slope = 1, intercept = 0, color = "red", linetype = "dashed") +
  labs(
    title = "Valores reales vs predichos",
    x = "Precio real (€)",
    y = "Precio predicho (€)"
  ) +
  theme_minimal()
```




```r
mape <- mean(abs((reales - predicciones) / reales)) * 100
cat("MAPE:", round(mape, 2), "%")
```

```
## MAPE: 23.38 %
```

Aunque el modelo muestra un buen ajuste general, el RMSE de 882 € puede considerarse elevado, especialmente en relación con los precios más frecuentes (1000–2000€).

Esto sugiere que el modelo funciona razonablemente bien para predicciones generales, pero pierde precisión en los extremos (especialmente en viviendas muy caras).

2. Regresión logística

2.1 Preparación de los datos.

```
datos <- datos %>%
  mutate(square_meters_price = price / square_meters)

datos <- datos %>%
  mutate(square_meters_price_re = ifelse(square_meters_price < 20, 0, 1))

datos$square_meters_price_re <- as.factor(datos$square_meters_price_re)

head(datos)
```

```
## # A tibble: 6 x 10
##   price rooms bathroom lift  terrace square_meters real_state neighborhood
##   <dbl> <dbl>    <dbl> <lgl> <lgl>           <dbl> <chr>      <chr>
## 1  3000     3        2 FALSE FALSE             150 apartment  Eixample
## 2  1250     1        1 FALSE FALSE              35 apartment  Eixample
## 3  2200     4        2 FALSE FALSE              90 apartment  Gracia
## 4  3468     3        2 FALSE FALSE              98 apartment  Ciutat Vella
## 5  1100     1        1 TRUE  FALSE              45 apartment  Les Corts
## 6  2500     3        2 TRUE  FALSE             180 apartment  Eixample
## # i 2 more variables: square_meters_price <dbl>, square_meters_price_re <fct>
```

```
datos_log <- datos %>%
  select(-price, -square_meters, -square_meters_price)

datos_log <- na.omit(datos_log)

set.seed(123)
n <- nrow(datos_log)
train_indices <- sample(seq_len(n), size = 0.8 * n)

datos_log_train <- datos_log[train_indices, ]
datos_log_test <- datos_log[-train_indices, ]

cat("Training:", nrow(datos_log_train), "observaciones\n")
```

```
## Training: 5883 observaciones
```

```
cat("Test:", nrow(datos_log_test), "observaciones\n")
```

```
## Test: 1471 observaciones
```

Se ha creado la variable square_meters_price, que representa el precio por metro cuadrado. A partir de esta, se ha generado square_meters_price_re, una variable dicotómica que toma el valor 1 si el precio por metro cuadrado es mayor o igual a 20€, y 0 si es inferior.

Para evitar colinealidad en el modelo de regresión logística, se han eliminado las variables price, square_meters y square_meters_price.

Finalmente, se ha dividido el conjunto de datos en entrenamiento (80%) y prueba (20%).

2.2 Estimación del modelo de regresión logística con el conjunto de entrenamiento

```
datos_log_train$lift <- as.factor(datos_log_train$lift)
datos_log_train$terrace <- as.factor(datos_log_train$terrace)
datos_log_train$real_state <- relevel(as.factor(datos_log_train$real_state), ref = "flat")
datos_log_train$neighborhood <- relevel(as.factor(datos_log_train$neighborhood), ref = "Ciutat Vella")

modelo_log <- glm(square_meters_price_re ~ ., data = datos_log_train, family = binomial)


summary(modelo_log)
```

```
##
## Call:
## glm(formula = square_meters_price_re ~ ., family = binomial,
##     data = datos_log_train)
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                    -0.62252    0.10950  -5.685 1.31e-08 ***
## rooms                          -0.74517    0.04501 -16.557  < 2e-16 ***
## bathroom                        0.55074    0.06310   8.728  < 2e-16 ***
## liftTRUE                       -0.41529    0.08421  -4.932 8.16e-07 ***
## terraceTRUE                     0.34748    0.09488   3.662 0.000250 ***
## real_stateapartment             2.12373    0.09543  22.255  < 2e-16 ***
## real_stateattic                 1.02837    0.15814   6.503 7.88e-11 ***
## real_statestudy                 0.47130    0.24616   1.915 0.055548 .
## neighborhoodEixample            0.36152    0.10208   3.542 0.000398 ***
## neighborhoodGracia             -0.21900    0.14682  -1.492 0.135810
## neighborhoodHorta- Guinardo    -1.85838    0.34687  -5.358 8.43e-08 ***
## neighborhoodLes Corts          -0.54491    0.18330  -2.973 0.002950 **
## neighborhoodNou Barris         -3.06245    1.01756  -3.010 0.002616 **
## neighborhoodSant Andreu        -1.57662    0.42532  -3.707 0.000210 ***
## neighborhoodSants-Montjuic     -0.41361    0.16389  -2.524 0.011615 *
## neighborhoodSarria-Sant Gervasi  0.11994    0.11967   1.002 0.316204
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6283.9  on 5882  degrees of freedom
## Residual deviance: 4887.4  on 5867  degrees of freedom
## AIC: 4919.4
##
## Number of Fisher Scoring iterations: 7
```

Los resultados muestran que las variables más influyentes en la probabilidad de que el precio por metro cuadrado sea mayor o igual a 20€ son rooms, bathroom, terrace, y el tipo de inmueble. Algunos barrios también presentan efectos significativos, como Eixample, Nou Barris o Les Corts.

En cambio, los barrios como Gracia, Sarria-Sant Gervasi y la variable real_statestudy no resultaron estadísticamente significativas, por lo que se podrían eliminar para simplificar el modelo sin pérdida de precisión.

2.3 Cálculo de las OR (Odss-Ratio)

```
or <- exp(coef(modelo_log))

or_ci <- exp(confint(modelo_log))
```

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

```
OR_tabla <- data.frame(
  OR = round(or, 3),
  CI_inf = round(or_ci[, 1], 3),
  CI_sup = round(or_ci[, 2], 3)
)

OR_tabla
```

```
##                                    OR CI_inf CI_sup
## (Intercept)                     0.537  0.433  0.665
## rooms                           0.475  0.434  0.518
## bathroom                        1.735  1.533  1.964
## liftTRUE                        0.660  0.560  0.779
## terraceTRUE                     1.416  1.174  1.703
## real_stateapartment             8.362  6.943 10.094
## real_stateattic                 2.796  2.047  3.807
## real_statestudy                 1.602  0.983  2.588
## neighborhoodEixample            1.436  1.176  1.755
## neighborhoodGracia              0.803  0.601  1.068
## neighborhoodHorta- Guinardo     0.156  0.074  0.293
## neighborhoodLes Corts           0.580  0.402  0.825
## neighborhoodNou Barris          0.047  0.003  0.218
## neighborhoodSant Andreu         0.207  0.082  0.445
## neighborhoodSants-Montjuic      0.661  0.477  0.907
## neighborhoodSarria-Sant Gervasi 1.127  0.891  1.425
```

Las variables con OR > 1, como bathroom, terrace, real_stateapartment, real_stateattic y neighborhood-Eixample, pueden considerarse factores de riesgo, ya que aumentan la probabilidad de que el precio por metro cuadrado sea mayor o igual a 20€.

Por otro lado, variables como rooms y lift presentan OR < 1, por lo que pueden considerarse factores de protección, al reducir dicha probabilidad.

Las variables real_statestudy y neighborhoodGracia no son concluyentes, ya que sus intervalos de confianza contienen el valor 1

2.4 Matriz de confusión

```
datos_log_test$lift <- as.factor(datos_log_test$lift)
datos_log_test$terrace <- as.factor(datos_log_test$terrace)
datos_log_test$real_state <- relevel(as.factor(datos_log_test$real_state), ref = "flat")
datos_log_test$neighborhood <- relevel(as.factor(datos_log_test$neighborhood), ref = "Ciutat Vella")
```

```r
probabilidades <- predict(modelo_log, newdata = datos_log_test, type = "response")

pred_clases <- ifelse(probabilidades >= 0.5, 1, 0)

reales <- as.numeric(as.character(datos_log_test$square_meters_price_re))

matriz_conf <- table(Predicho = pred_clases, Real = reales)
matriz_conf
```

```
##         Real
## Predicho    0    1
##        0 1067  222
##        1   53  129
```

```r
VP <- matriz_conf["1", "1"]  # Verdaderos Positivos
VN <- matriz_conf["0", "0"]  # Verdaderos Negativos
FP <- matriz_conf["1", "0"]  # Falsos Positivos
FN <- matriz_conf["0", "1"]  # Falsos Negativos

sensibilidad <- VP / (VP + FN)
especificidad <- VN / (VN + FP)

cat("Sensibilidad:", round(sensibilidad, 3), "\n")
```

```
## Sensibilidad: 0.368
```

```r
cat("Especificidad:", round(especificidad, 3), "\n")
```

```
## Especificidad: 0.953
```

La matriz de confusión muestra que el modelo:

- Clasificó correctamente 1067 pisos baratos (Verdaderos Negativos)
- Clasificó correctamente 129 pisos caros (Verdaderos Positivos)
- Se equivocó con 222 pisos caros (Falsos Negativos)
- Y con 53 pisos baratos (Falsos Positivos)

Las métricas clave son:

- Sensibilidad = 36.89%; el modelo detecta pocos pisos caros, por lo que tiene baja sensibilidad.
- Especificidad = 95.27%; el modelo distingue muy bien los pisos baratos.

Esto sugiere que el modelo prioriza la precisión en detectar pisos baratos, pero tiene dificultad para identificar correctamente los pisos con precio alto por metro cuadrado.

2.5 Predicción

```r
datos_resultados <- datos_log_test
datos_resultados$probabilidad <- round(probabilidades, 4)

viviendas_caras <- datos_resultados[datos_resultados$probabilidad >= 0.5, ]

viviendas_caras_tabla <- viviendas_caras[, c("neighborhood", "real_state", "probabilidad")]

head(viviendas_caras_tabla, 10)
```

```
## # A tibble: 10 x 3
##    neighborhood   real_state probabilidad
##    <fct>          <fct>             <dbl>
##  1 Ciutat Vella   apartment         0.591
##  2 Ciutat Vella   apartment         0.591
##  3 Eixample       apartment         0.814
##  4 Sants-Montjuic apartment         0.710
##  5 Ciutat Vella   apartment         0.787
##  6 Eixample       apartment         0.578
##  7 Ciutat Vella   apartment         0.837
##  8 Ciutat Vella   apartment         0.709
##  9 Ciutat Vella   apartment         0.668
## 10 Ciutat Vella   apartment         0.621
```

Se han identificado las viviendas del conjunto de prueba cuya probabilidad de tener un precio por metro cuadrado superior o igual a 20€ es mayor o igual a 0.5, según el modelo de regresión logística ajustado.

La mayoría de las viviendas clasificadas como caras pertenecen a los barrios de Ciutat Vella, Eixample y Sants-Montjuic, y son de tipo apartment.

Esto es coherente con los resultados obtenidos en los apartados anteriores, donde estos barrios y este tipo de vivienda mostraron una mayor probabilidad de pertenecer a la clase cara.

2.6 Bondad del ajuste y curva ROC

```r
dev_null <- modelo_log$null.deviance
dev_residual <- modelo_log$deviance
df_diff <- modelo_log$df.null - modelo_log$df.residual

chi_valor <- dev_null - dev_residual

p_valor <- pchisq(chi_valor, df = df_diff, lower.tail = FALSE)

cat("Chi-cuadrado:", round(chi_valor, 3), "\n")
```

```
## Chi-cuadrado: 1396.46
```

```r
cat("p-valor:", round(p_valor, 5), "\n")
```

```
## p-valor: 0
```

```r
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.

##
## Adjuntando el paquete: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

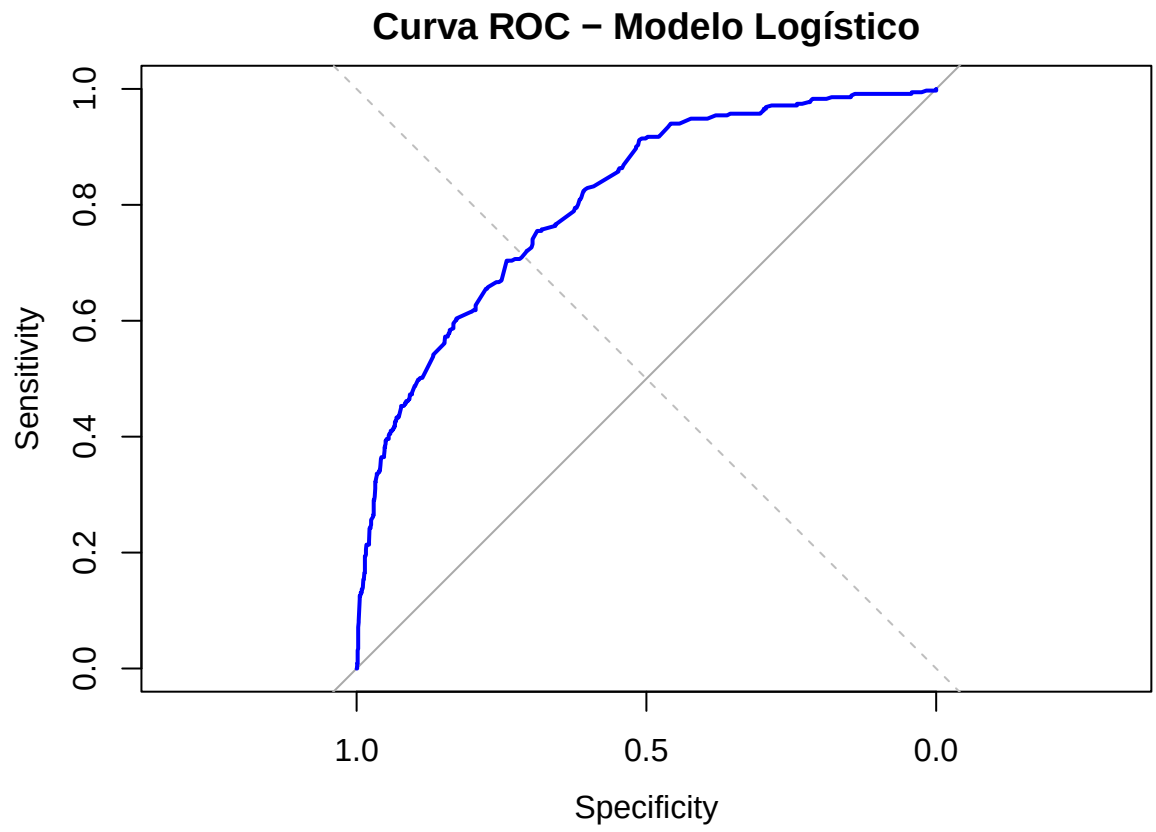
```r
roc_obj <- roc(datos_log_test$square_meters_price_re, probabilidades)
```

```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```

```r
auc_valor <- auc(roc_obj)
cat("AUC:", round(auc_valor, 3), "\n")
```

```
## AUC: 0.807
```

```r
plot(roc_obj, col = "blue", main = "Curva ROC - Modelo Logístico")
abline(a = 0, b = 1, lty = 2, col = "gray")
```



Se ha evaluado la bondad del ajuste del modelo logístico mediante:

1. Test Chi-cuadrado:

- La diferencia entre la devianza nula y la residual ha sido de 1396.46, con un p-valor = 0, lo que indica que el modelo mejora significativamente al modelo sin predictores.

2. Curva ROC y AUC:

- La curva ROC muestra una buena capacidad de discriminación.
- El valor del AUC ha sido de 0.807, lo que implica que el modelo tiene una probabilidad del 80.7% de clasificar correctamente una vivienda como cara o barata.

En conjunto, estos resultados confirman que el modelo es estadísticamente significativo y útil para la clasificación, aunque, como se observó anteriormente, podría mejorarse la sensibilidad (detección de viviendas caras).

2.7 Modelo de regresión logística multinomial

```
library(nnet)

datos$real_state <- as.factor(datos$real_state)

datos_multi <- datos[, c("real_state", "price", "rooms", "bathroom", "square_meters")]
datos_multi <- na.omit(datos_multi)

set.seed(123)
n <- nrow(datos_multi)
train_indices <- sample(1:n, size = 0.8 * n)

multi_train <- datos_multi[train_indices, ]
multi_test <- datos_multi[-train_indices, ]

modelo_multi <- multinom(real_state ~ price + rooms + bathroom + square_meters, data = multi_train)
```

```
## # weights:  24 (15 variable)
## initial  value 8815.445842
## iter  10 value 3769.662655
## iter  20 value 3248.571072
## iter  30 value 3177.001382
## iter  40 value 3176.894154
## final  value 3176.847289
## converged
```

```
pred_multi <- predict(modelo_multi, newdata = multi_test)

accuracy <- mean(pred_multi == multi_test$real_state)
cat("Precisión del modelo multinomial:", round(accuracy * 100, 2), "%\n")
```

```
## Precisión del modelo multinomial: 83.71 %
```

```r
table(Predicho = pred_multi, Real = multi_test$real_state)
```

```
##            Real
## Predicho    apartment attic flat study
##   apartment        42     3   13     0
##   attic             1     0    1     0
##   flat            151    62 1282    24
##   study             1     0    3     7
```

Se ha creado un modelo de regresión logística multinomial para clasificar el tipo de vivienda (real_state) en función del precio, número de habitaciones, baños y metros cuadrados.

El modelo alcanzó una precisión global del 83.71%, lo cual refleja un buen desempeño general. Sin embargo, según la matriz de confusión, el modelo tiende a clasificar la mayoría de las viviendas como flat, debido a la alta frecuencia de esta clase en el conjunto de datos. Este sesgo hacia la clase mayoritaria implica que el modelo predice bien las viviendas flat, pero tiene dificultades para identificar correctamente attic, apartment y study.

Para mejorar la clasificación en las clases minoritarias, podría considerarse balancear los datos, o aplicar modelos más robustos como árboles de decisión o random forest.

3 Resumen ejecutivo. Conclusiones del análisis

```r
library(knitr)

resumen <- data.frame(
  Apartado = c("1.1", "1.2", "1.3", "1.4", "1.5", "1.6", "1.7",
               "2.1-2.2", "2.3", "2.4", "2.5", "2.6", "2.7"),
  Pregunta = c(
    "¿Qué variables cuantitativas se relacionan con el precio?",
    "¿Qué barrios y tipos de vivienda tienen precios más altos?",
    "¿Cómo se dividen los datos en entrenamiento y test?",
    "¿Qué variables explican el precio con un modelo lineal?",
    "¿Mejora el modelo al añadir variables cualitativas?",
    "¿Cumple el modelo los supuestos de regresión lineal?",
    "¿Qué tan bien predice el modelo el precio?",
    "¿Qué factores explican si un piso es 'caro' (> 20€/m²)?",
    "¿Qué variables son factores de riesgo o protección?",
    "¿Qué tal clasifica el modelo logístico binario?",
    "¿Qué viviendas se predicen como caras?",
    "¿El modelo es estadísticamente válido y discriminativo?",
    "¿Se puede predecir el tipo de vivienda?"
  ),
  Resultado = c(
    "Mayor correlación: square_meters (0.69), bathroom (0.58)",
    "Barrios más caros: Sarria-Sant Gervasi, Eixample",
    "Train: 5883, Test: 1471 (80/20)",
    "R² ajustado = 0.5134; todas significativas",
    "R² ajustado = 0.5878; mejora sin colinealidad",
    "Residuos aceptables; ligera heterocedasticidad",
    "RMSE = 882.61 €, MAPE = 23.38%",
    "Variables clave: bathroom, terrace, tipo y barrio",
```

```
    "OR > 1: riesgo; OR < 1: protección (ej. rooms, lift)",
    "Sensibilidad: 36.9%; Especificidad: 95.3%",
    "Mayoría: Ciutat Vella, Eixample, tipo apartment",
    "Chi² = 1396.46 (p = 0); AUC = 0.807",
    "Precisión = 83.71%; buen resultado pero sesgo a 'flat'"
  ),
  Conclusión = c(
    "El tamaño y número de baños son claves para el precio",
    "El barrio influye fuertemente en el precio de alquiler",
    "División adecuada para modelado supervisado",
    "El modelo explica bien el precio con variables cuantitativas",
    "Mejora al añadir factores cualitativos; se mantiene estabilidad",
    "El modelo cumple los supuestos de regresión lineal",
    "El modelo funciona bien, aunque con margen de mejora en extremos",
    "Modelo logístico binario útil; factores claramente influyentes",
    "Algunos factores aumentan o reducen la probabilidad de ser 'caro'",
    "Detecta muy bien pisos baratos, peor los caros",
    "Zonas caras detectadas correctamente según el modelo",
    "El modelo tiene valor explicativo y buena discriminación",
    "Buen rendimiento general; necesita mejorar minorías"
  )
)

kable(resumen, format = "markdown", align = "l")
```

Apartado	Pregunta	Resultado	Conclusión
1.1	¿Qué variables cuantitativas se relacionan con el precio?	Mayor correlación: square_meters (0.69), bathroom (0.58)	El tamaño y número de baños son claves para el precio
1.2	¿Qué barrios y tipos de vivienda tienen precios más altos?	Barrios más caros: Sarria-Sant Gervasi, Eixample	El barrio influye fuertemente en el precio de alquiler
1.3	¿Cómo se dividen los datos en entrenamiento y test?	Train: 5883, Test: 1471 (80/20)	División adecuada para modelado supervisado
1.4	¿Qué variables explican el precio con un modelo lineal?	R^2 ajustado = 0.5134; todas significativas	El modelo explica bien el precio con variables cuantitativas
1.5	¿Mejora el modelo al añadir variables cualitativas?	R^2 ajustado = 0.5878; mejora sin colinealidad	Mejora al añadir factores cualitativos; se mantiene estabilidad
1.6	¿Cumple el modelo los supuestos de regresión lineal?	Residuos aceptables; ligera heterocedasticidad	El modelo cumple los supuestos de regresión lineal
1.7	¿Qué tan bien predice el modelo el precio?	RMSE = 882.61 €, MAPE = 23.38%	El modelo funciona bien, aunque con margen de mejora en extremos
2.1–2.2	¿Qué factores explican si un piso es 'caro' (> 20€/m^2)?	Variables clave: bathroom, terrace, tipo y barrio	Modelo logístico binario útil; factores claramente influyentes
2.3	¿Qué variables son factores de riesgo o protección?	OR > 1: riesgo; OR < 1: protección (ej. rooms, lift)	Algunos factores aumentan o reducen la probabilidad de ser 'caro'
2.4	¿Qué tal clasifica el modelo logístico binario?	Sensibilidad: 36.9%; Especificidad: 95.3%	Detecta muy bien pisos baratos, peor los caros

Apartado	Pregunta	Resultado	Conclusión
2.5	¿Qué viviendas se predicen como caras?	Mayoría: Ciutat Vella, Eixample, tipo apartment	Zonas caras detectadas correctamente según el modelo
2.6	¿El modelo es estadísticamente válido y discriminativo?	$Chi^2 = 1396.46$ ($p = 0$); AUC = 0.807	El modelo tiene valor explicativo y buena discriminación
2.7	¿Se puede predecir el tipo de vivienda?	Precisión = 83.71%; buen resultado pero sesgo a 'flat'	Buen rendimiento general; necesita mejorar minorías