Practica 2

Félix Mucha & Jose Enriquez

2023-06-11

Contents

Descripción del dataset

El dataset contiene información de venta de departamentos de la ciudad de Buenos Aires, dicha información nos va a permitir realizar análisis, investigación, predicción y toma de decisiones para la selección del mejor inmueble. Las variables del dataset son:

- created_on: fecha de publicación de aviso
- operation: operación de venta
- property_type: tipo de propiedad
- place_with_parent_names: lugar de la propiedad
- lat.lon: Latitud y longitud

- lat: Latitud
- lon: Longitud
- price_aprox_usd: Precio aprox en usd
- surface_total_in_m2: superficie_total_en_m2
- surface_covered_in_m2: superficie_cubierta_en_m2
- price_usd_per_m2: precio_usd_por_m2
- floor: piso
- rooms: habitaciones
- expenses: gastos
- properati_url: URL_propiedad
- barrio: barrio
- comuna: comuna

Origende datos Kaggle (https://www.kaggle.com/datasets/gastonmichelotti/properati-data-set).

Con esta información buscamos detectar las comunas que presentan un alto precio aproximado ($) de los inmuebles. Esto nos va a permitir generar y tomar decisiones basadas en el comportamiento de los inmuebles a nivel georáfico y la influencia del número de habitaciones en los precios. Además, evaluar si existe alguna variación de los precios con respecto al m2. Asimismo, buscamos ajustar un modelo de regresión lineal que nos permita predecir los precios aprox. de los inmuebles.

Objetivo

- Determinar si el modelo de regresión lineal múltiple es el adecuado para predecir los precios aproximados de los inmuebles.
- Determinar los puntos geográficos de los inmuebles de mayor ínteres.

Carga de datos

```
# Carga de datos
data <- read.csv("../data/datos_properati.csv", stringsAsFactors = FALSE, encoding='utf-8')

# Revisión de las variables
names(data)
```

```
##  [1] "created_on"              "operation"
##  [3] "property_type"           "place_with_parent_names"
##  [5] "lat.lon"                 "lat"
##  [7] "lon"                     "price_aprox_usd"
##  [9] "surface_total_in_m2"     "surface_covered_in_m2"
## [11] "price_usd_per_m2"        "floor"
## [13] "rooms"                   "expenses"
## [15] "properati_url"           "barrio"
## [17] "comuna"
```

La data está compuesta por 17 variables y 18 979 registros de diferentes inmuebles en la ciudad de Buenos Aires.

Integración y selección

Para el caso de estudio conservaremos la mayoría de las variables, para realizar el análisis respectivo. Entre las variables que se van a retirar tenemos la *operation*, *place_with_parent_names* y la *properati_url*. La

variable *operation* esta compuesta por una sola categoría, lo cuál no aporta información relevante. Asimismo, la variable *place_with_parent_names* a pesar de mostrar la ubicación de los inmuebles tenemos la latitud y longitud para tener la ubicación más precisa de los inmuebles, además tenemos información del *barrio* y la *comuna* para brindar mayor información a la ubicación. Luego de la selección de variables, analizaremos el tipo de dato de cada variable.

```
##            created_on         property_type              lat.lon
##           "character"           "character"            "character"
##                   lat                   lon       price_aprox_usd
##             "numeric"             "numeric"              "numeric"
##   surface_total_in_m2 surface_covered_in_m2      price_usd_per_m2
##             "numeric"             "numeric"              "numeric"
##                 floor                 rooms               expenses
##             "numeric"             "numeric"              "numeric"
##         properati_url                barrio                 comuna
##           "character"           "character"              "numeric"
```

Con respecto al tipo de dato, tenemos que convertir *created_on* en formato fecha. Además, es necesario convertir a factor las variables *property_type*, *rooms*, *barrio* y *comuna*. Por otro lado, dejaremos la variable *lat.lon* como un tipo de character.

```
##            created_on         property_type              lat.lon
##                "Date"              "factor"            "character"
##                   lat                   lon       price_aprox_usd
##             "numeric"             "numeric"              "numeric"
##   surface_total_in_m2 surface_covered_in_m2      price_usd_per_m2
##             "numeric"             "numeric"              "numeric"
##                 floor                 rooms               expenses
##             "numeric"             "integer"              "numeric"
##         properati_url                barrio                 comuna
##           "character"              "factor"               "factor"
```

De este resumen estadístico podemos obtener información relevante de cada variable, dependiento del tipo de variable. Además, nos permite apreciar el comportamiento de las variables numéricas con respecto a la media y sus valores extremos. De la misma forma, podemos detectar posibles comportamientos anómalos como la variable *floor* que hay un inmueble con 904 pisos, para validar la información es necesario usar el *properati_url*.

Limpieza de los datos

Lo primero que realizaremos es calcular el procentaje de nulos que existen por cada variable de interes del análisis.

```
#calculando el procentaje de nulos por columna
porcentaje_nulos <- calcular_porcentaje_nulos(datos)
porcentaje_nulos
```

```
##   created_on property_type lat.lon lat lon price_aprox_usd surface_total_in_m2
## 1          0             0       0   0   0        8.082618            12.60867
##   surface_covered_in_m2 price_usd_per_m2    floor    rooms expenses
## 1              11.08067         15.50134 85.47869 28.77918 79.89884
##   properati_url barrio comuna
## 1             0      0      0
```

Nos vamos a enfocarnos sólo en la venta de departamentos para tener datos comparables e imputables.

```
# Filtramos solo departamentos
Dpto <- filter(datos, property_type == 'apartment')

porc_null_dpto <- calcular_porcentaje_nulos(Dpto)
porc_null_dpto
```

```
##   created_on property_type lat.lon lat lon price_aprox_usd surface_total_in_m2
## 1          0             0       0   0   0        7.293848            10.17762
##   surface_covered_in_m2 price_usd_per_m2    floor    rooms expenses
## 1              9.731884         12.96684 83.24441 22.48936 75.93706
##   properati_url barrio comuna
## 1             0      0      0
```

De la proporción podemos ver que en las variables *floor* y *expenses*, la proporción de nulos es muy elevada (null $> 70\%$). Por este motivo, estás variables serán retiradas del análisis. Asimismo, se retirará *property_type* ya que solo trabajaremos con la categoría **apartmet**

```
# Eliminación de variables
dfDpto <- select(Dpto, -floor, -expenses, -property_type)

# Nueva proporción de nulos
calcular_porcentaje_nulos(dfDpto)
```

```
##   created_on lat.lon lat lon price_aprox_usd surface_total_in_m2
## 1          0       0   0   0        7.293848            10.17762
##   surface_covered_in_m2 price_usd_per_m2    rooms properati_url barrio comuna
## 1              9.731884         12.96684 22.48936             0      0      0
```

Observamos que una variable importante es el número de habitaciones se obsera que hay 22.5% de nulos para el análisis no vamos a considerar los nulos.

```
# Filtra los valores no nulos
dfDptoNN <- dfDpto %>% filter(complete.cases(.))

porc_null_dptoN <- calcular_porcentaje_nulos(dfDptoNN)
porc_null_dptoN
```

```
##   created_on lat.lon lat lon price_aprox_usd surface_total_in_m2
## 1          0       0   0   0               0                   0
##   surface_covered_in_m2 price_usd_per_m2 rooms properati_url barrio comuna
## 1                     0               0     0             0      0      0
```

La variable *surface_total_in_m2* y *surface_covered_in_m2* estan relacionados. Esto se debe a que en teoría la superficie total (m2) es mayor a la superficie cubierta (m2). Por ello, validaremos que se cumpla este requisito.

```
# Validación del requisito
dfDptoNN$val <- dfDptoNN$surface_total_in_m2 >= dfDptoNN$surface_covered_in_m2
dfDptoNN <- dfDptoNN[dfDptoNN$val == 'TRUE',]
dfDptoNN <- select(dfDptoNN, -val)
```

La variable *surface_total_in_m2* será limitada a trabajar solo con los departamentos inferiores a los 200 m2 y que los precios por m2 sean inferiores a 6000$.

```
# Filtro de datos
dfDptoN <- dfDptoNN[(dfDptoNN$surface_total_in_m2 <= 200) & (dfDptoNN$price_usd_per_m2 <= 6000),]
```

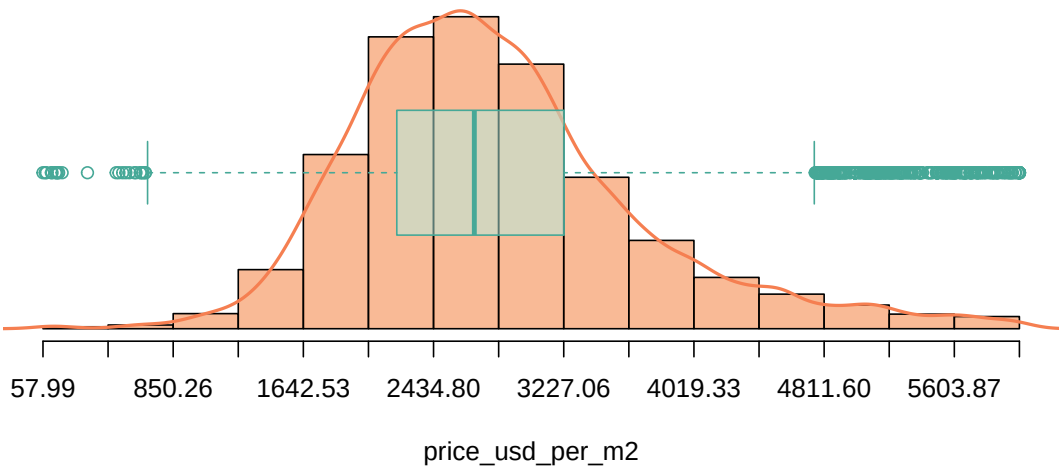
A continuación, analizaremos el comportamiento de la variable *rooms*.

```
##
##    1    2    3    4    5    6    7    8    9   10   11   13   30
## 2017 2529 2544 1646  366   95   24    6    1    1    1    1    1
```

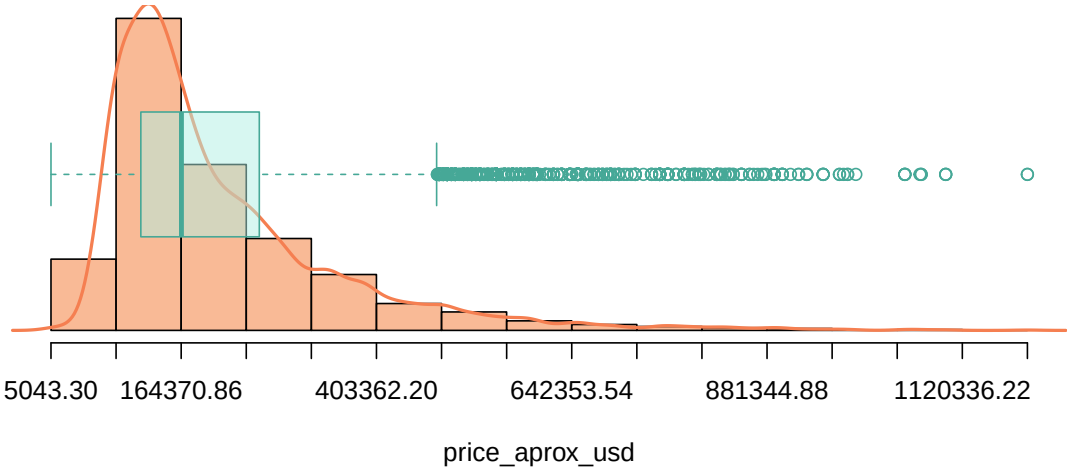
De los siguientes resultados obtenidos, llegamos a la conclusión de que es necesario agrupar los valores mayor a 6 habitaciones para continuar trabajando correctamente.

```
# Agrupación de valores
dfDptoN$rooms <- ifelse(dfDptoN$rooms < 6, dfDptoN$rooms, 6)
```

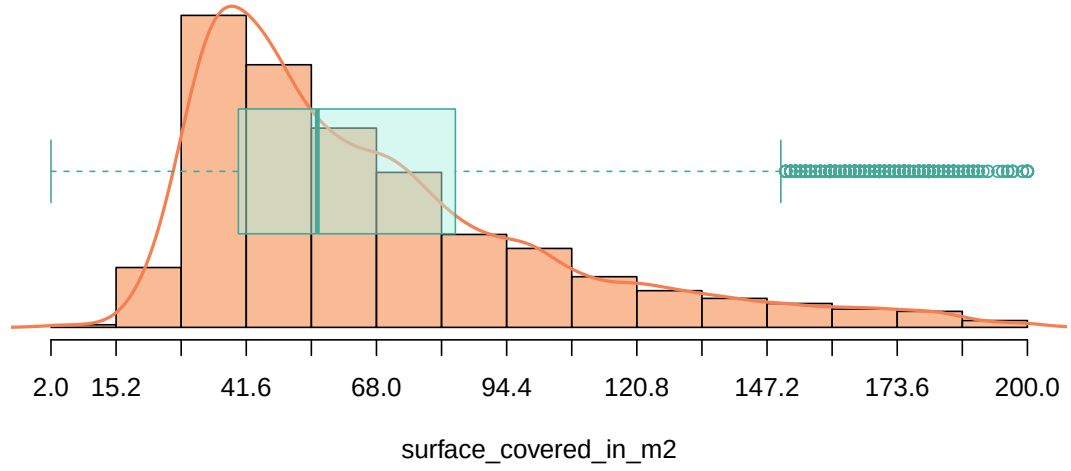
En los gráficos siguientes analizaremos el comportamiento de las variables y la existencia de observaciones atípicas. Así como también analizaremos si las variables presentan una distribución simétrica o asimétrica.



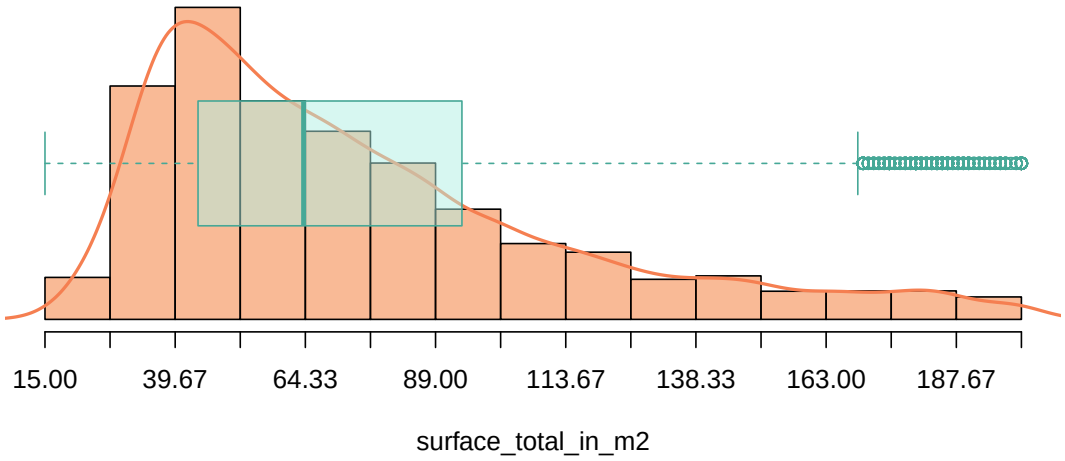
Del gráfico, podemos apreciar la existencia de observaciones con un comportamiento atípico, estas observaciones se encuentran situados en la cola izquierda, ya que, el precio por m2 están por debajo de límite inferior ($802). Asimismo, la distribución de los datos tienden a ser simérticos con respecto a la mediana (mediana: 2641.65), pero la existencia de observaciones distorcionan su comportamiento. Por este motivo, es necesario realizar el gráfico *qqplot* y la prueba de *Anderson Darling* para analizar si el comportamiento de los datos se ajustan a una distribución normal.



En el siguiente gráfico, podemos observar la existencia de observaciones con un comportamiento atípico, estas observaciones se encuentran situados a la derecha del diagrama de cajas, ya que, el precio aproximado están por encima del límite superior ($399408). Asimismo, la distribución de los datos tienden a ser asimérticos, ya que la mayor concentración de datos se encuentran entre los 5043 y los 251521. Asimismo, se puede apreciar que la diferencia entre los valores de la media y mediana son grandes (media: 187491; mediana: 159000). Es así que los comportamientos anómalos distorcionan la distribución de los datos.



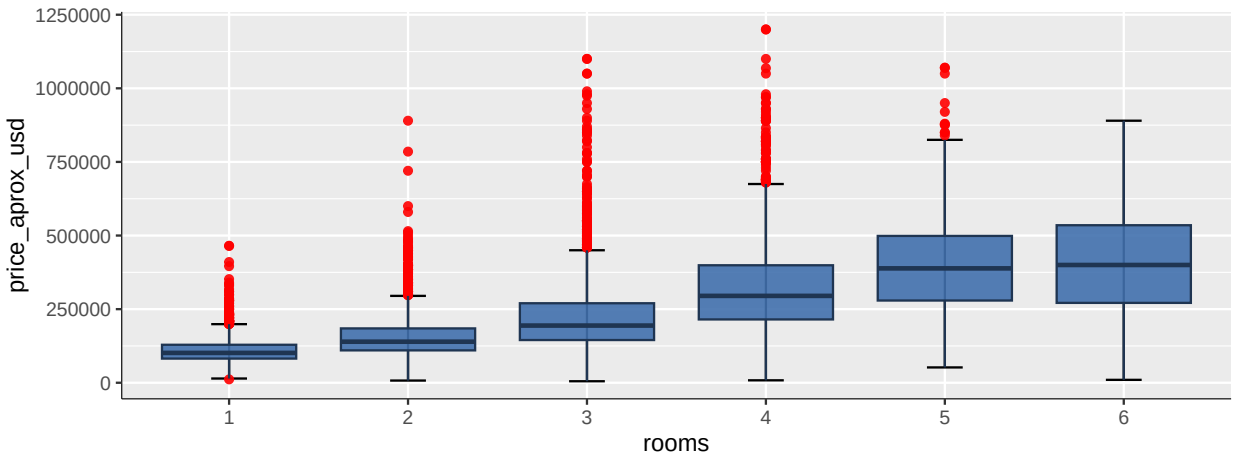
Con respecto al gráfico de la superficie cubierta (m2), podemos observar la existencia de observaciones con un comportamiento atípico, estas observaciones se encuentran situados a la derecha del diagrama de cajas, ya que, la superficie cubierta está por encima del límite superior (159 m2). Asimismo, la distribución de los datos tienden a ser asimérticos, ya que la mayor concentración de datos se encuentran entre los 16 m2 y los 81 m2. Asimismo, se puede apreciar que la diferencia entre los valores de la media y mediana son grandes (media: 56 m2; mediana: 67 m2). Es así que los comportamientos anómalos distorcionan la distribución de los datos.



Por último, el gráfico de la superficie total (m2), esta muy correlacionado con el comportamiento la superficie cubierta (m2). Por este motivo, es que la distribución y la existencia de datos anómalos son similares al de la gráfica anterior.

Analizamos la evolución de los precios, en función al número de habitaciones:

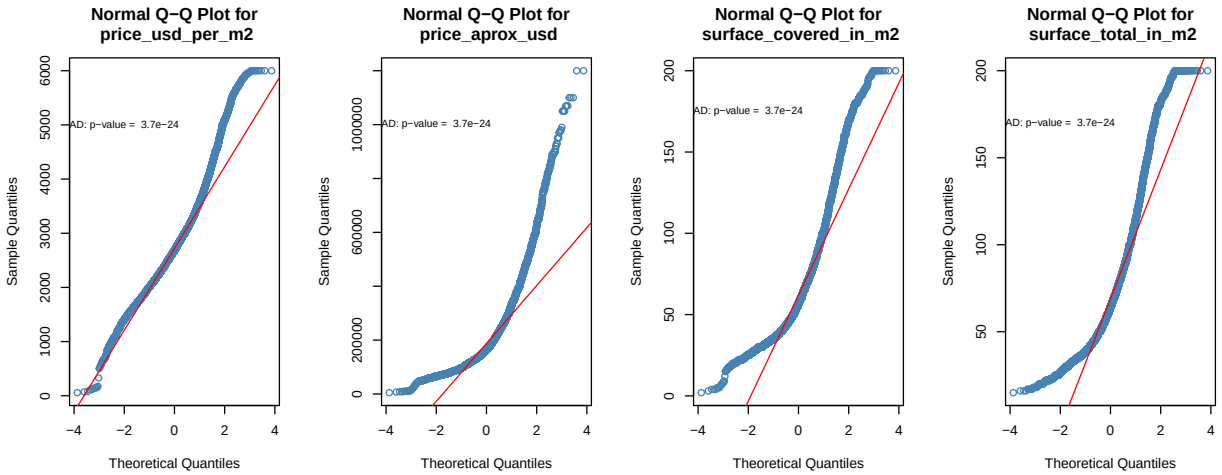
En este primer análisis, evaluaremos el comportamiento de los precios en función al número de habitaciones.



En este primer gráfico, podemos observar que a mayor número de habitaciones la media de los precios estimados de un inmueble tienden a elevarse. Asimismo, si el inmueble cuenta con 5 o más habitaciones la media del precio estimado está por encima de los $350 000. Además, si los inmuebles cuentan con 5 o más habitaciones, la variación del precio es mínima. Con esto quiero decir, si un inmueble cuenta con más de 5 habitaciones, el precio es definido por otros factores.

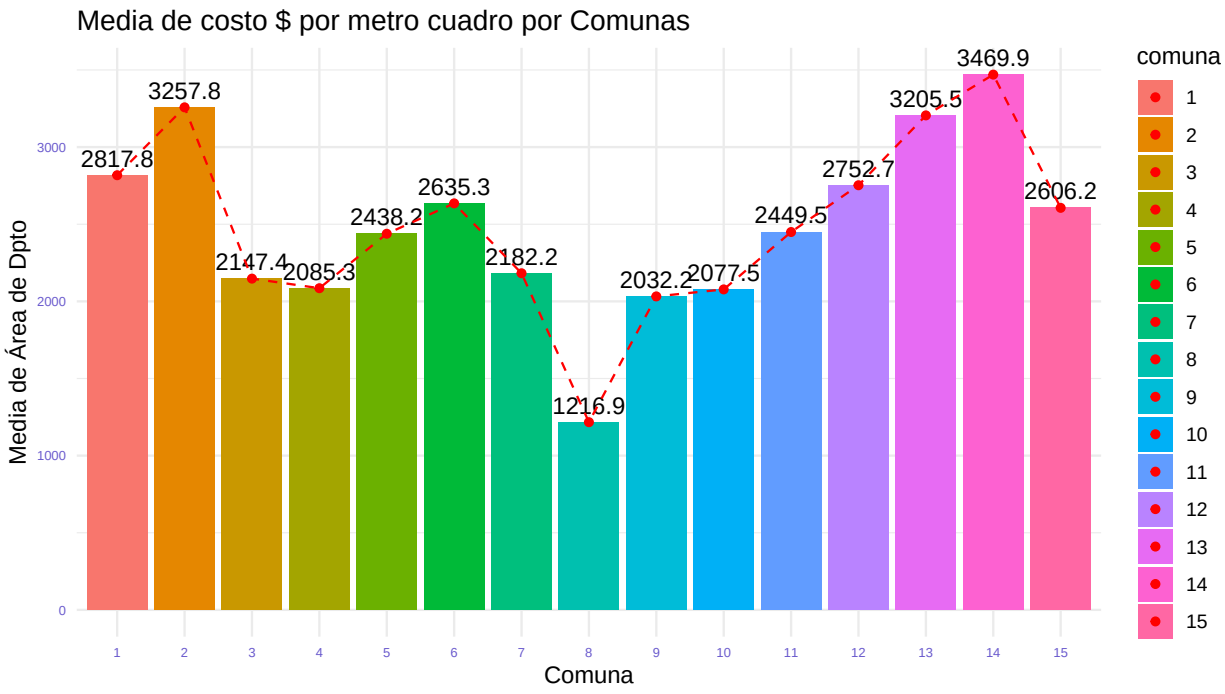
Por otro lado, existen inmuebles que tienen un menor número de habitaciones y el precio aproximado esta muy por encima de la media de los precios. Es así que se puede conlcuir que el precio de estos inmuebles están definidos por otros factores como la ubicación geografica.

Revisión de la distribución de los datos

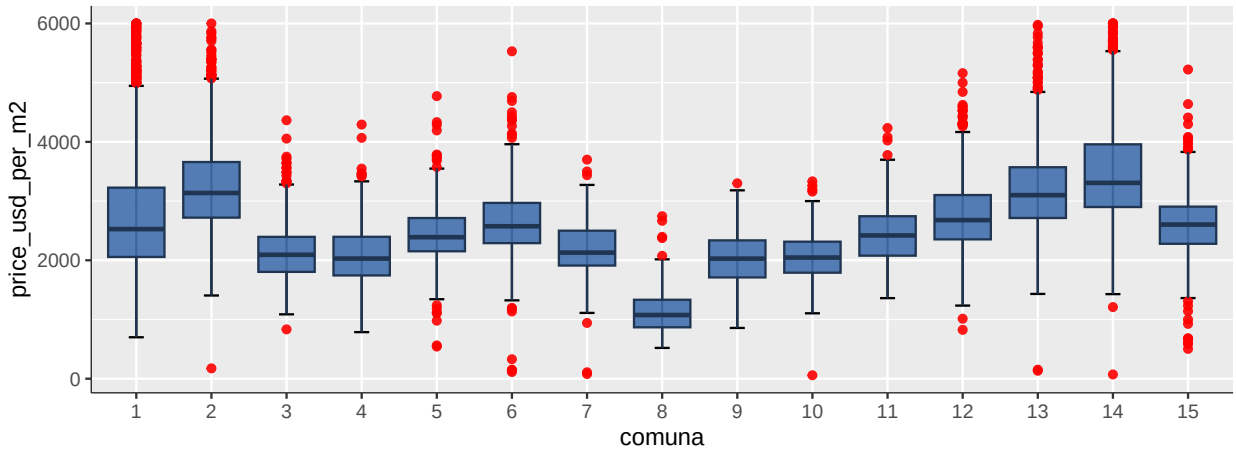


El test nos indica que ninguna de las variables se ajusta a una distribución normal, ya que el p-valor es inferior al coeficiente 0.05, por lo que hay suficiente evidencia estadística para rechazar la hipotesis nula y esto da a entender que los datos no se ajusta a una distribución normal.

Analizamos la evolución de los precios, en función a las comunas:

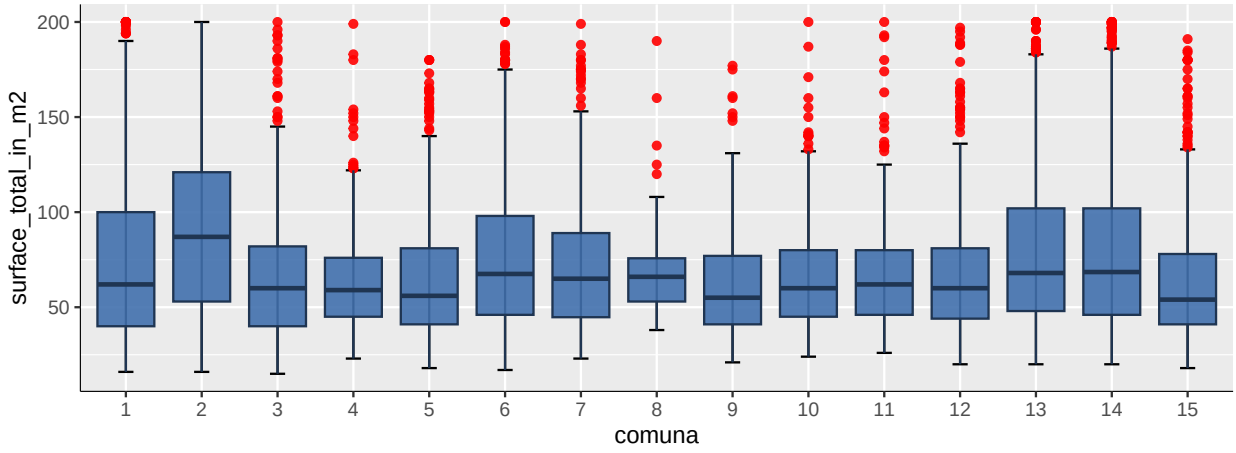


Analizamos el comportamiento del precio por m2, en las comunas



Del siguiente gráfico, podemos concluir que la comuna 2, 14 y considerando la comuna 13 tienen los precios más elevados, ya que la media de los precios por m2 están por encima de los $3000. Por otro lado, se puede apreciar que la comuna 8 tiene los inmuebles con el menor precio por m2. Asimismo, la mayoría de las comunas tiene una media de precio por m2 que se encuentran entre los 2000 usd y 3000 usd.

Analizamos el comportamiento de la superficie total (m2), en las comunas



Los datos atípicos los vamos a sustituir con nulos para luego, imputarlos por valores más reales en la sigueinte sección.

```
# Replicamos la data
dfDptoNN <- dfDptoN

# Trataremos los valores atípicos
quantiles <- quantile(dfDptoNN$price_aprox_usd, c(0.05, 0.95), na.rm = TRUE)
dfDptoNN[dfDptoNN$price_aprox_usd>quantiles[2], "price_aprox_usd"]<-NA
summary(dfDptoNN$price_aprox_usd)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    5043  113563  159000  187491  240000  498000     461
```

Luego de imputar los valores atípicos observamos el resumen para ver efectivamente que se realizo la imputación.

```
quantiles <- quantile(dfDptoNN$surface_total_in_m2, c(0.05, 0.95), na.rm = TRUE)
dfDptoNN[dfDptoNN$surface_total_in_m2>quantiles[2], "surface_total_in_m2"]<-NA
summary(dfDptoNN$surface_total_in_m2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   15.00   44.00   62.00   69.27   88.00  163.00     460
```

```
quantiles <- quantile(dfDptoNN$price_usd_per_m2, c(0.05, 0.95), na.rm = TRUE)
dfDptoNN[dfDptoNN$price_usd_per_m2>quantiles[2], "price_usd_per_m2"]<-NA
summary(dfDptoNN$price_usd_per_m2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   57.99 2184.38 2641.65 2680.76 3130.07 4523.81     462
```

Imputación de datos

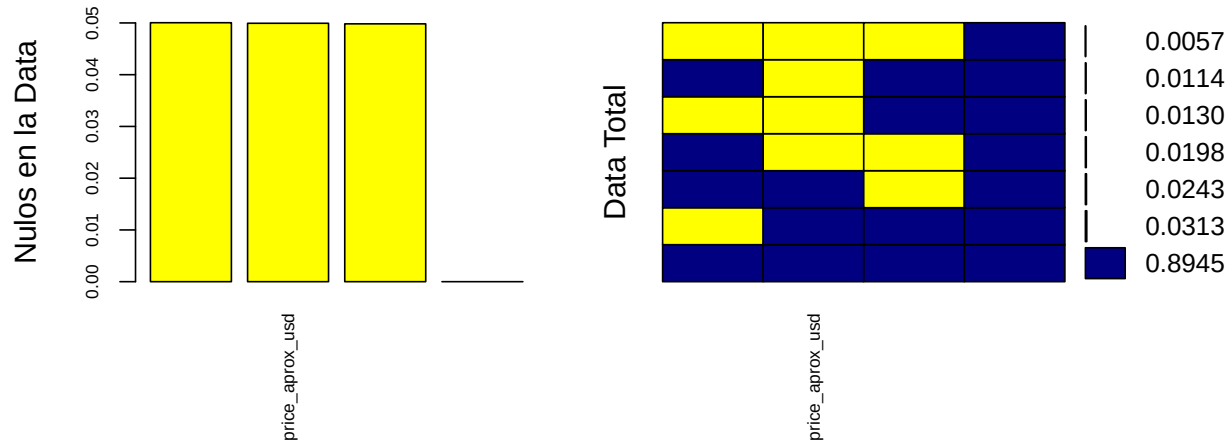
Normalización de datos.- Se crea una función para normalizar las variables de la información de los inmubeles que estamos analizando.

```
##   price_aprox_usd surface_total_in_m2 surface_covered_in_m2 price_usd_per_m2
## 1       0.6794850           0.7770270             0.7329193        0.5726592
## 2       0.4401131           0.2500000             0.2857143        0.9429939
## 3       0.5374847           0.3513514             0.3850932        0.8893911
```

Creamos un modelo de entranamiento y testeo.

En la siguiente reusmen podremos observar los valores reales y los que han predecidos con el modelo.

Antes de realizar impuatción de valores observaremos un resumen de los datos a imoputar.




```
##
##  Variables sorted by number of missings:
##                Variable      Count
##       price_usd_per_m2 0.05004333
##        price_aprox_usd 0.04993501
##     surface_total_in_m2 0.04982669
##   surface_covered_in_m2 0.00000000
```

Este gráfico nos muestra que el 89% de la información no tien nulos, lo cuál nos favorece porque estamos trabajando con información buena.

Análisis

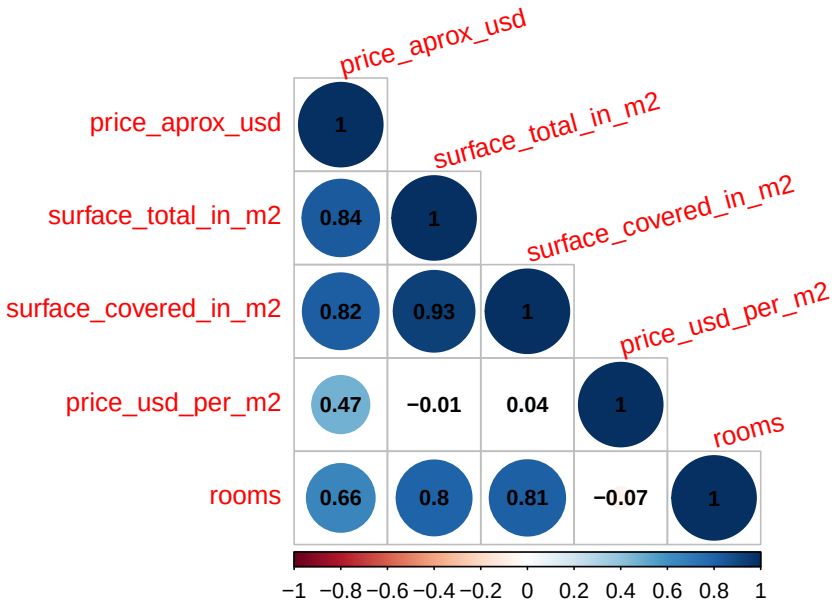
En el análisis evaluaremos los datos para poder realizar un modelamiento predictivo, en la estimación del precio aproximado ($). En un primer avance seleccionaremos las posibles variables que expliquen la variación de los precios.

```
df_model <- dplyr::select(input_data_mapa, -created_on, -lat.lon, -lat, -lon,
              -properati_url, -price_aprox_usd_imp, -surface_total_in_m2_imp,
              -price_usd_per_m2_imp, -long)
head(df_model, 3)
```

```
##   price_aprox_usd surface_total_in_m2 surface_covered_in_m2 price_usd_per_m2
## 1          340000                 130                  120         2615.385
## 2          222000                  52                   48         4269.231
## 3          270000                  67                   64         4029.851
##   rooms  barrio comuna
## 1     4 PALERMO     14
## 2     1 PALERMO     14
## 3     3 PALERMO     14
```

A continuación, examinaremos la correlación entre las variables, esto nos va a permitir identificar variables que puedan elevar la colinealidad entre variables.

```
# Analizaremos la coorelación de variables
correlacion <- round(cor(df_model[,c('price_aprox_usd', 'surface_total_in_m2', 'surface_covered_in_m2',
```



De la siguiente tabla podemos ver que la superficie total (m2) y la superficie cubierta por los usuarios (m2) están correlacionados linealmente en un 93%, es por ello, que la suérficie cubierta será retirada del análisis.

Asimismo, el número de habitaciones presenta una alta correlación lineal positiva con la supericie total (m2) y la superficie cubierta (m2) en un 80% y 81%, respectivamente. Por ello, el número de habitaciones será retirada del análisis.

```
# Eliminación de variables correlacionadas
df_modelc <- dplyr::select(df_model, -rooms, -surface_covered_in_m2)
head(df_modelc, 3)
```

```
##   price_aprox_usd surface_total_in_m2 price_usd_per_m2  barrio comuna
## 1          340000                 130         2615.385 PALERMO     14
## 2          222000                  52         4269.231 PALERMO     14
## 3          270000                  67         4029.851 PALERMO     14
```

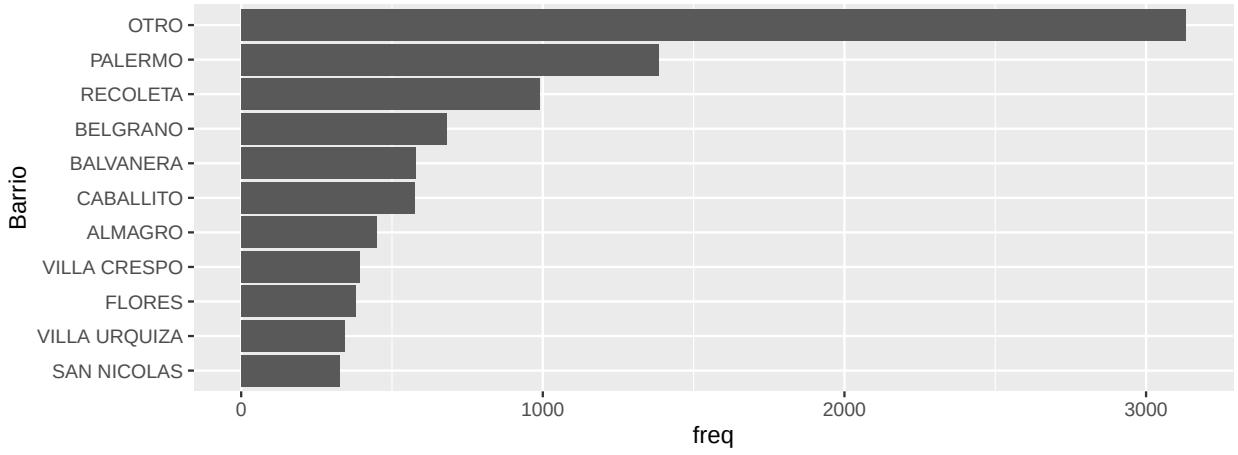
Ahora veremos las categorías de las variables de tipo cualitativas.

Como se puede mostrar en la gráfica, es necesario agrupar categorías, con el objetivo de reducir el número de categorías en las variables y además de tener categorías más consistentes. ya que hay barrios que registran 9 o 4 inmuebles. Es por ello, que es preferible reducir as categorías. Graficamente podemos asignar el punto de corte de 300, este valor referencial nos va a permitir agrupar todas las categorías que se encuentren por debajo de los 300. A este grupo se le denominará como otro.

Luego, reemplazamos dichas categorías como *OTRO* barrio.

```
# Reemplazamo la categoría
df_modelc$n_barrio <- df_modelc$barrio
df_modelc$n_barrio <- as.character(df_modelc$n_barrio)
df_modelc['n_barrio'] <- lapply(df_modelc['n_barrio'], function(x) replace(x,x %in% name_b_otro, 'OTRO'
```

Finalmente, las nuevas categorías de la variable n_barrio quedan clasificadas correctamente.



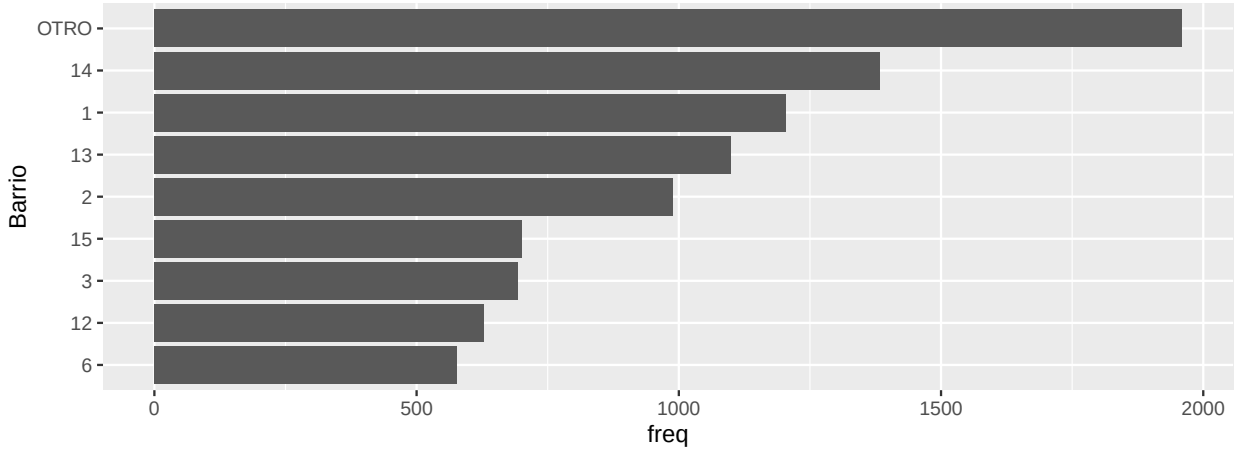
Replicaremos el mismo procedimiento con la comuna.

En este caso el punto de referencial para el corte es de 550.

Luego, reemplazamos dichas categorías como *OTRO* comuna.

```
# Reemplazamo la categoría
df_modelc$n_comuna <- df_modelc$comuna
df_modelc$n_comuna <- as.character(df_modelc$n_comuna)
df_modelc['n_comuna'] <- lapply(df_modelc['n_comuna'], function(x) replace(x,x %in% name_c_otro, 'OTRO'
```

Finalmente, las nuevas categorías de la variable n_comuna quedan clasificadas correctamente.



Por ende, una vez codificados correctamente las categorías. Es necesario eliminar las anteriores.

```
# Eliminación de variables
df_model_pre <- dplyr::select(df_modelc, -barrio, -comuna)
head(df_model_pre, 3)
```

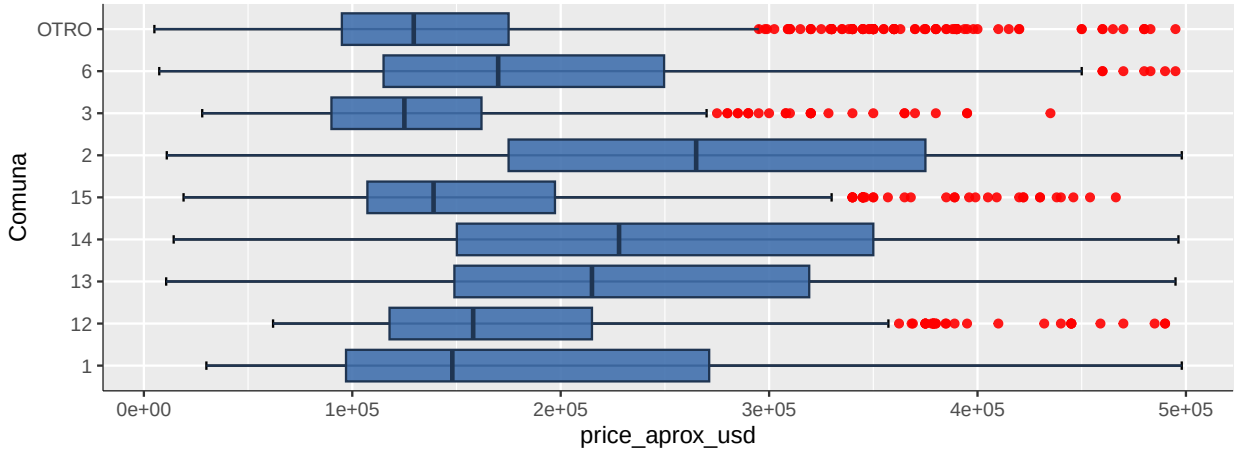
```
##   price_aprox_usd surface_total_in_m2 price_usd_per_m2 n_barrio n_comuna
## 1          340000                 130         2615.385  PALERMO       14
## 2          222000                  52         4269.231  PALERMO       14
## 3          270000                  67         4029.851  PALERMO       14
```

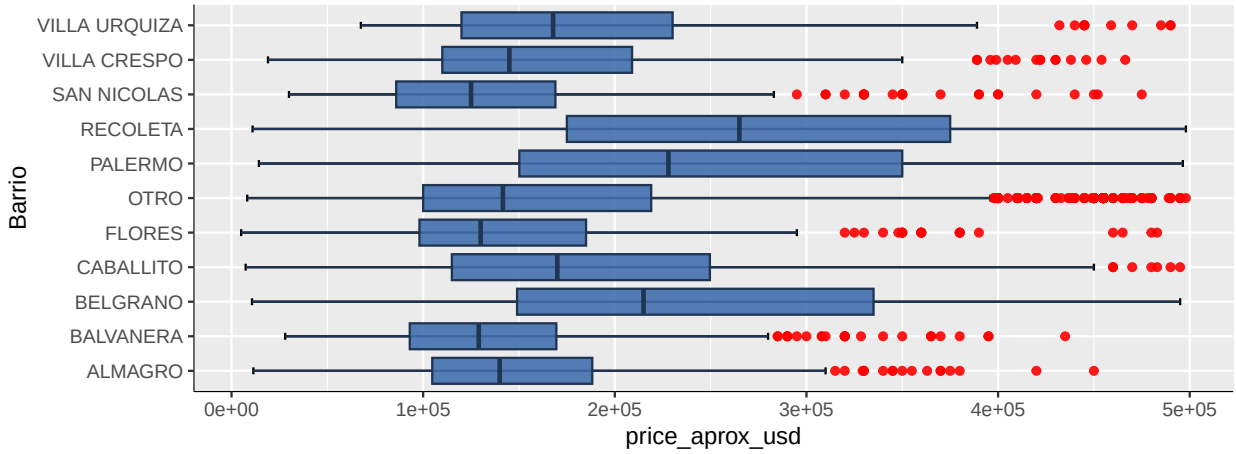
Ahora evaluaremos si existe una relación entre las variables *n_barrio* y *n_comuna*, usanla el test de independencia.

```
# Test de independencia
tabla <- table(df_model_pre$n_barrio, df_model_pre$n_comuna)
chisq.test(tabla)
```

```
##
##  Pearson's Chi-squared test
##
## data:  tabla
## X-squared = 53654, df = 80, p-value < 2.2e-16
```

Esta prueba nos indica que con un nivel de significancia del 5%, hay suficiente evidencia estadística para reachazar la hipótesis nula. Es decir, hay suficiente evidencia estadística para asegurar que el barrio y la comuna están relacionados significativamente. A continuación, veremos la importancia de las variables con respecto al precio aproximado ($), para definir que variable es más importante.





De los siguientes gráficos, podemos apreciar que existe una mayor visibilidad del comportamiento del los barrios sobre los precios aproximados. Por ello, la variable comuna será retirada del análisis.

```
dfmodelamiento <- dplyr::select(df_model_pre, -n_comuna)
head(dfmodelamiento, 3)
```

```
##   price_aprox_usd surface_total_in_m2 price_usd_per_m2 n_barrio
## 1          340000                 130         2615.385  PALERMO
## 2          222000                  52         4269.231  PALERMO
## 3          270000                  67         4029.851  PALERMO
```

Modelamiento predictivo

Luego del análisis de los datos, pasaremos a ajustar un modelo de regresión lineal múltiple para poder explicar el precio aproximado de los inmuebles. Pero antes realizaremos una partición de los datos parasu validación.

```
set.seed(1998)
indice <- caret::createDataPartition(dfmodelamiento$price_aprox_usd,times = 1, p = 0.85, list = F)

# Dimensión de la data de entrenamiento
dftrain <- dfmodelamiento[indice,]
dim(dftrain)
```

```
## [1] 7848    4
```

```
# Dimensión de la data de testeo
dftest <- dfmodelamiento[-indice,]
dim(dftest)
```

```
## [1] 1384    4
```

```
ModelF <- lm(price_aprox_usd ~ surface_total_in_m2 + price_usd_per_m2 + n_barrio , data = dftrain)
summary(ModelF)
```

```
##
## Call:
## lm(formula = price_aprox_usd ~ surface_total_in_m2 + price_usd_per_m2 +
##     n_barrio, data = dftrain)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -202414  -11068    -598    9951  201360
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -1.991e+05  2.067e+03 -96.316  < 2e-16 ***
## surface_total_in_m2    2.706e+03  9.413e+00 287.525  < 2e-16 ***
## price_usd_per_m2       7.185e+01  5.080e-01 141.447  < 2e-16 ***
## n_barrioBALVANERA      9.961e+02  1.973e+03   0.505 0.613761
## n_barrioBELGRANO       6.362e+03  1.926e+03   3.303 0.000960 ***
## n_barrioCABALLITO      3.191e+03  1.957e+03   1.631 0.102935
## n_barrioFLORES         3.828e+03  2.181e+03   1.755 0.079267 .
## n_barrioOTRO           7.001e+03  1.565e+03   4.474 7.77e-06 ***
## n_barrioPALERMO        6.087e+03  1.746e+03   3.486 0.000493 ***
## n_barrioRECOLETA       8.636e+03  1.812e+03   4.767 1.91e-06 ***
## n_barrioSAN NICOLAS    2.537e+03  2.258e+03   1.124 0.261080
## n_barrioVILLA CRESPO   4.380e+03  2.145e+03   2.042 0.041213 *
## n_barrioVILLA URQUIZA  2.335e+03  2.247e+03   1.039 0.298924
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 28780 on 7835 degrees of freedom
## Multiple R-squared:  0.9357, Adjusted R-squared:  0.9356
## F-statistic:  9499 on 12 and 7835 DF,  p-value: < 2.2e-16
```

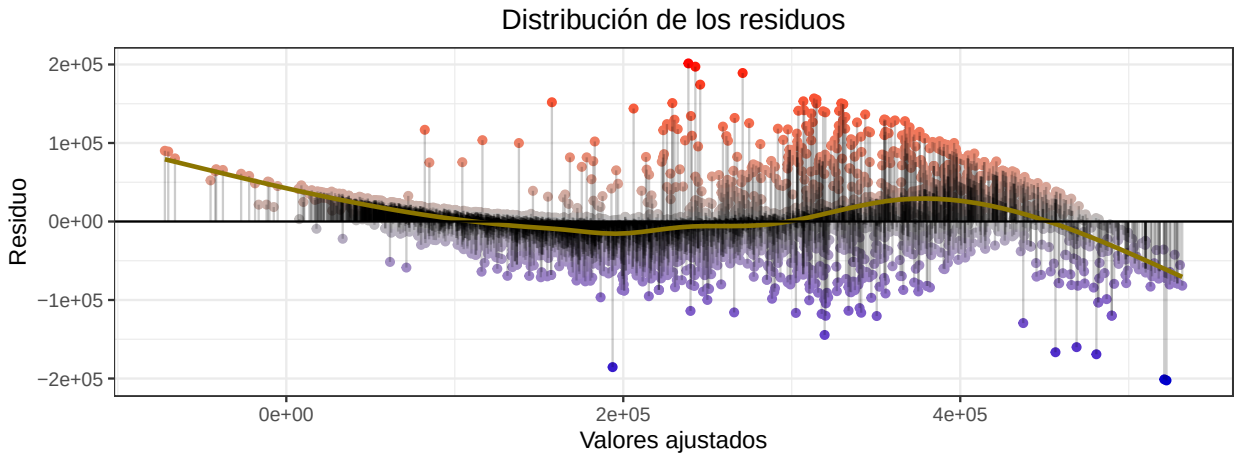
Evaluamos la existencia de multicolinealidad

```
car::vif(ModelF)
```

```
##                         GVIF Df GVIF^(1/(2*Df))
## surface_total_in_m2 1.051506  1        1.025429
## price_usd_per_m2    1.360145  1        1.166253
## n_barrio            1.414979 10        1.017507
```

Otra forma de medir la colinealidad entre las variables independientes del modelo es mediante el análisis del Factor de Inflación de la Varianza (VIF). Si los resultados obtenidos se encuentran entre 1 y 5, esto indica una correlación moderada entre las variables, pero pueden ser controlados. Por este motivo, se puede concluir la existencia de colinealidad entre las variables, pero estás pueden ser controladas debidamente.

- Analizaremos si la variabilidad es constante (Homocedasticidad)



Gráficamente podemos observar que las observaciones de los residuos y los valores ajustados, siguen un ligero patrón. Por este motivo no se puede asegurar la aleatoriedad de los datos, es decir, la variabilidad de los datos no son constantes. Para, verificar si la variabilidad es o no constante usaremos la prueba de Breusch-Pagan con un nivel de significancia del 5%, determinaremos la prueba de hipótesis.

$H0 : Los\ \ errores\ \ tienen\ \ varianza\ \ constante.$

$H1 : Los\ \ errores\ \ no\ \ tienen\ \ varianza\ \ constante.$

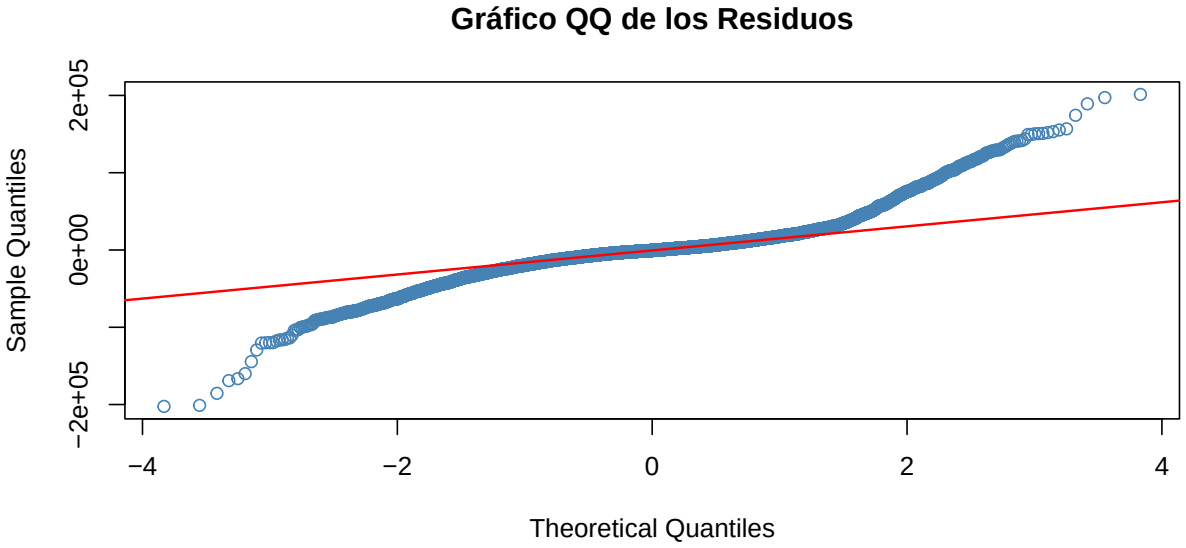
```
bptest(ModelF)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  ModelF
## BP = 707, df = 12, p-value < 2.2e-16
```

De la prueba Breusch-Pagan podemos concluir, con un nivel de significancia del 5% hay suficiente evidencia estadística para rechazar la hipótesis nula. Es decir, La variabilidad de los errores no es constante, por ello, no se cumple con el supuesto de homocedasticidad de los residuos.

- Analizaremos si la distribución de los residuos se ajustan a una normal

Para analizar la distribución de los residuos es necesario realizar un gráfico QQ-plot, este nos permitirá deducir si los residuos se ajustan a una distribución normal.



Del siguiente gráfico podemos inferir que los residuos no se ajustan a una distribución normal. Esto se debe a que la mayoría de los datos no están superpuestos en la recta. Sin embargo, hay grupos de datos que están alejados de la recta lo que indica que los datos tienen una ligera asimérica en la derecha e izquierda (colas pesadas). Para verificar que los residuos se ajustan o no a la normalidad de los datos usaremos la prueba de *Anderson - Darling*.

$H0$: *Los errores siguen una distribución normal.*

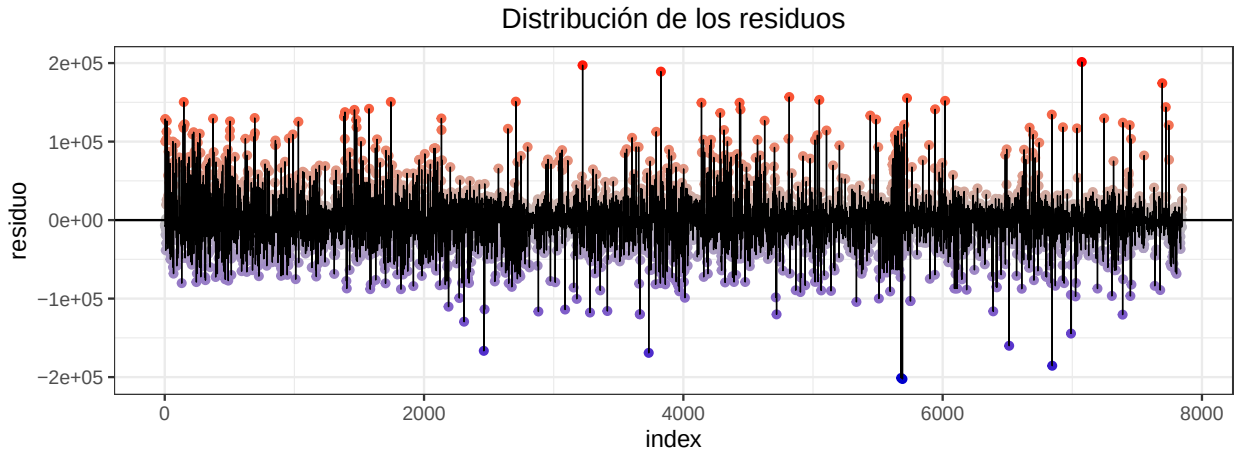
$H1$: *Los errores no se ajustan a una distribución normal.*

```
ad.test(residuo)
```

```
##
##  Anderson-Darling normality test
##
## data:  residuo
## A = 250.7, p-value < 2.2e-16
```

Del test de normalidad podemos concluir, con un nivel de significancia del 5% hay suficiente evidencia para rechazar la hipótesis nula. Por este motivo, los errores no se ajustan a una distribución normal, lo que implica que no cumple con la condición de la normalidad de los residuos.

Otro supuesto que debemos analizar es la autocorrelación de residuos.



Graficamente, podemos deducir que existe una ligera tendencia que nos puede hacer sospechar de la existencia de la autocorrelación de residuos.

Aplicamos el test de Durbin-Watson, para comprobar si existe autocorrelación de residuos.

H0: No existe autocorrelación en los residuos.

H1: Existe autocorrelación en los residuos.

```
dwtest(ModelF)
```

```
##
##  Durbin-Watson test
##
## data:  ModelF
## DW = 1.8929, p-value = 5.81e-07
## alternative hypothesis: true autocorrelation is greater than 0
```

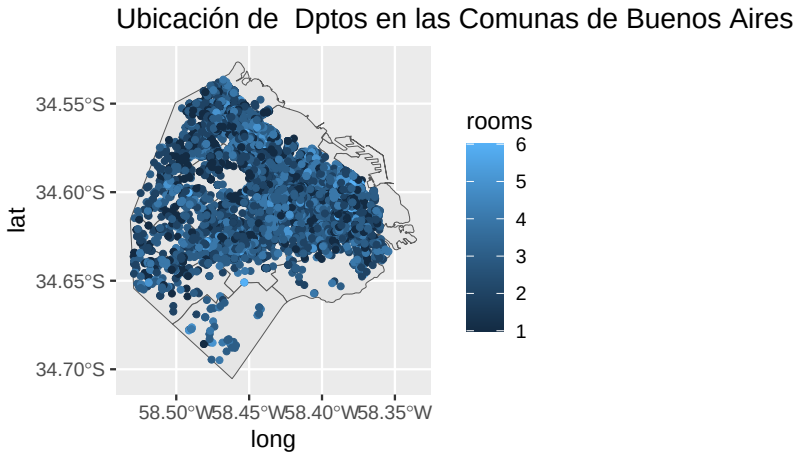
Con un nivel de significancia del 5%, concluimos que hay evidencia estadística para rechazar la hipótesis nula. Por lo tanto, los errores del modelo están autocorrelacionados.

Representación de resultados

Representación geográfica de los resultados

Después de imputar los datos podemos mostralo para que los compradores puedan ubicarse más facilemente la propeidad que buscan.

```
## Reading layer ‘comunas’ from data source
##   ‘D:\Master UOC\Clases\Primer Semestre\TipologÃa y ciclo de vida de los datos\PrÃ¡ctica2\Final04\P2
##   using driver ‘GeoJSON’
## Simple feature collection with 15 features and 6 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -58.53152 ymin: -34.70529 xmax: -58.33515 ymax: -34.52649
## Geodetic CRS:  WGS 84
```



Al observar las 15 comunidades juntas no es muy legible la ubicación de las propiedades por ello sólo observaremos de una comunidad.

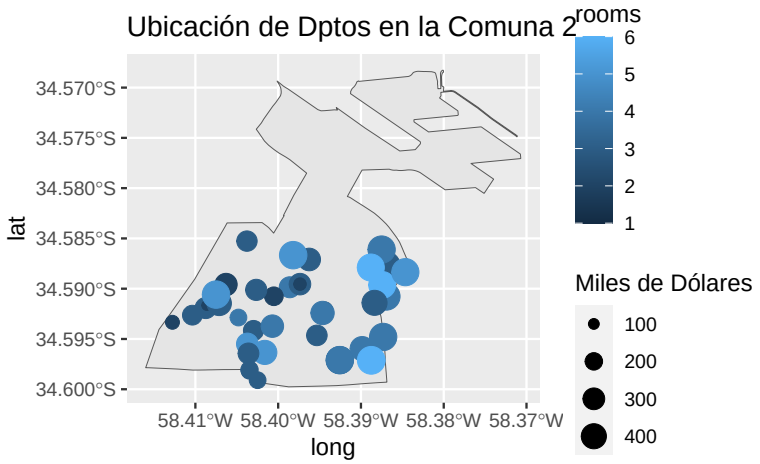
En el siguiente gráfico extraeremos sólo los primeros 50 departamentos de la comunidad 2 y lo gráficaremos.

```
# Para una mejor visualización de la ubicación sólo grafiacremos las tres primeras comunidades.
data_mapa_comuna02 <- filter(input_data_mapa, comuna == 2)
```

Observamos un resumen de la cantidad de deparatmento por número de habitaciones.

```
##
##   1   2   3   4   5   6
## 163 186 255 273  76  35
```

En el siguiente gráfico extraeremos sólo los primeros 50 departamentos de la comunidad 2 y lo gráficaremos.



Con este ejemplo graficamos sólo una proción de la información con lo cuál se puede mostrar que se puede generar la ubicación de los inmubeles de ínteres teniendo algunso criterios como: por número de deprata-mento, por precio de departamento y área que se busca.

Resolución del problema y conclusiones

El modelo de regresión lineal múltiple no es considerada como una buena opción en este proyecto. Esto se debe a que no cumple con los supuestos necesarios para poder tener una consistencia en el modelo. Es así

que vez que los errores no se ajustan a una distribución normal (p_value<0.05). Al igual que los supuesto de homocedasticidad y heterocedasticidad no cumplen con los suficientes requisitos para que puedan satisfacer dichos supuestos. Por este motivo, llegamos a la conclusión de que aplicar una regresión lineal múltiple en estos datos no es una opción viable, a menos que se puedan solucionar el problema con los supuestos.

Por otro lado, una vía para poder continuar el análisis es mediante la categorización de los precios aproximados y convertir el modelo de regresión en un modelo de clasificación. Esta transformación podría solucionar probablemente algunas métricas al igual que los supuestos.

Exportación del código en R y de los datos producidos

El código en R esta incluido en este fichero con extensión rmd y tambien se puede descargar en GitHub desde la siguiente dirección:

https://github.com/JoseC468/P2-procesamiento-datos/tree/main/data

Los datos de salida se exportan mediante el siguiente comando y pueden ser descargados desde en GitHub desde la siguiente dirección:

```
write.csv(input_data_mapa, file = "../data/datos_properati_out.csv")
```

Tabla de contribuciones

Contribuciones	Firma 1	Firma 2
Investigación previa	Jose	Félix
Redacción de las respuestas	Jose	Félix
Desarrollo del código	Jose	Félix
Participación en el video	Jose	Félix