MEMORIA PROYECTO FINAL

1. Introducción

Partimos del dataset facilitado sobre los datos de Airbnb con el que nos planteamos unas suposiciones iniciales que fueron:

1. El precio y el barrio tenían relación.
2. La imagen era relevante a la hora de alquilar.
3. El precio del alquiler influye en el número de reseñas.

A partir de ellas comenzamos nuestras labores con el proyecto de limpieza, muestreo, visualización de datos, verificación de la fiabilidad de los datos obtenidos y los cruces realizados.

2. Dataset y limpieza de datos

Hemos descargado el dataset y lo abrimos en pandas para procesar el dataframe; con el column.values sacamos el listado entero de los nombres de las columnas para saber con qué datos estamos trabajando y si hay que modificar algún nombre para la correcta lectura y procesamiento.

Se modifican los nombres de las columnas para normalizarlas. Se confirma si hay duplicados para eliminar, no se encuentran duplicados. Con group by vemos muchos outliers.

El campo Features incluye información concatenada de algunos atributos del alojamiento. Esa información se desagrega creando ocho campos booleanos para esos ocho atributos utilizando la función str.contains de Pandas.

Se ha filtrado por zipcode solo dejando los que comienzan por 28 que serían los de Madrid con str.startswith y se quedan en 12876 filas. Dejamos todas las columnas a eliminar en el mismo chunk. No elimino la métrica de fotos para hacerlo booleano y trabajar con él.

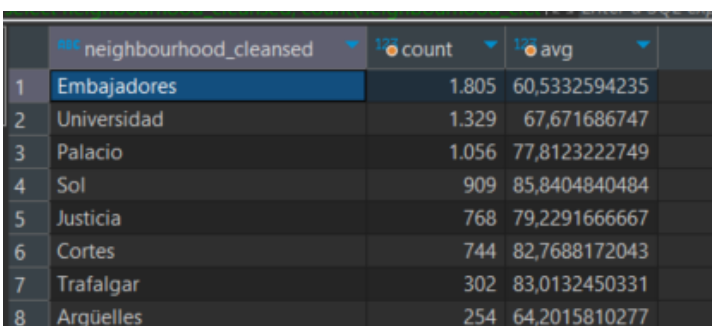
Hacemos un segundo campo calculado para cambiar los NaN a 0. Hago un gráfico de caja con los datos calculados, comparando los dos campos calculados. Veo muchísimos outliers en el boxplot de los NaN a 0.

<u>Datawarehouse</u>

Cargamos el CSV con las 31 columnas en DBeaver. Cambiamos el tipo de datos. Confirmamos en Dbeaver que “neighbourhood cleansed” está limpio y apto para usar.

Con SQL vemos los 10 barrios con mayores alquileres. Guardo este dato para comparar al hacer visualizaciones.

Comparamos los precios promedios de los alquileres de los 10 barrios más alquilados, el precio promedio de cada alquiler cuando el host es superhost es más alto.

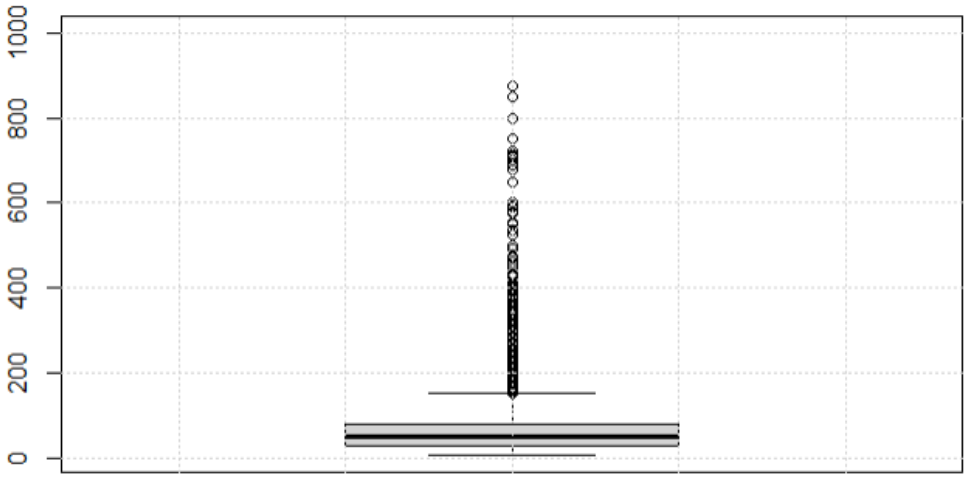


	neighbourhood_cleansed	count	avg
1	Embajadores	1.805	60,5332594235
2	Universidad	1.329	67,671686747
3	Palacio	1.056	77,8123222749
4	Sol	909	85,8404840484
5	Justicia	768	79,2291666667
6	Cortes	744	82,7688172043
7	Trafalgar	302	83,0132450331
8	Argüelles	254	64,2015810277

A	B	C	D	E	F
superhost False	Superhost True	avg		SuperHost False	Superhost True
58,8931	71,5923	60,5333		-2,71%	18,27%
66,4467	75,7429	67,6717		-1,81%	11,93%
75,8197	88,7963	77,8123		-2,56%	14,12%
84,9656	91,4309	85,8405		-1,02%	6,51%
79,1465	79,7453	79,2292		-0,10%	0,65%
81,4954	91,7935	82,7688		-1,54%	10,90%
84,0037	44,7000	83,0132		1,19%	-46,15%
59,4742	89,3750	64,2016		-7,36%	39,21%
54,9289	68,3824	53,2988		3,06%	28,30%
88,6851	74,6563	85,4744		3,76%	-12,66%

3. Análisis exploratorio estadístico.

Boxplot de los precios de alquiler filtrados por madrid:

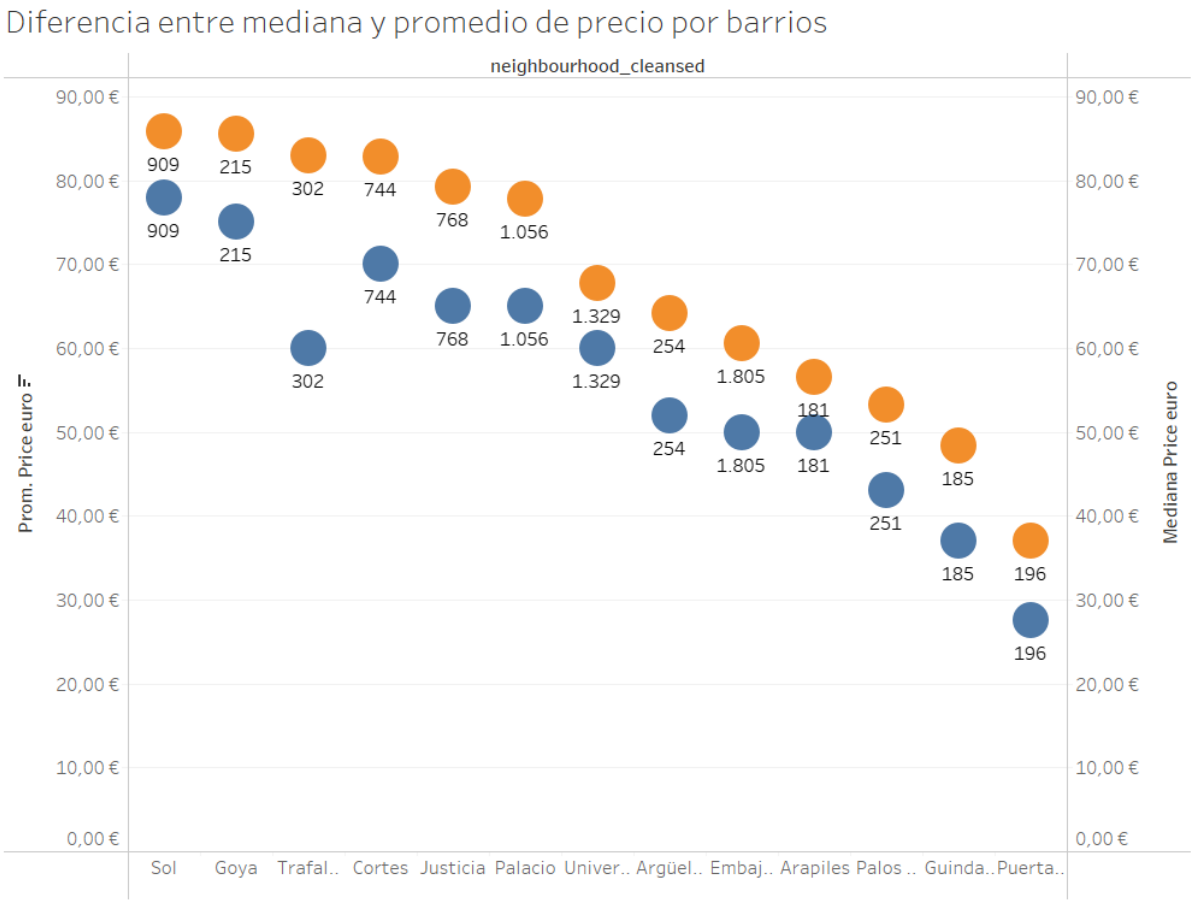


Vemos que hay bastantes outliers por encima del precio de 200€ incluso algo menos.

4. Visualización métricas

¿Qué factores influyen en el alquiler de habitaciones/apartamentos en Madrid? Métricas: Availability 365; Foto → Picture Url (si hay true si no false); Precio → Price; Precio calculado → Security Deposit and Cleaning Fee and Price; Number of Reviews; Review Score Values; Neighborhood/Zip Code/; Host Since; Cancellation Policy.

Hay 126 barrios, y 20 grupos de barrios. Hacemos dos gráficos comparando promedio y media de precios según los 13 barrios con más alquileres. 13 para que sea un 10% de muestra. Hacemos otro gráfico con una muestra aleatoria de barrios.



La diferencia entre media y mediana es constante (obviando el outlier) por más que la diferencia en cantidad de alquileres sea grande. Significa que tenemos siempre un alquiler que levanta la media.

5. Modelo de regresión

La tarea asignada es hacer un algoritmo de regresión lineal que prediga el precio de un inmueble en función de las características que elijáis.

Recomendación de precio a los host. Primero se ha cargado el data frame limpio con las 12.872 filas y 31 columnas a Rstudio. Ahí se han ido analizando las columnas y la correlación de estas con el precio.

Para empezar a crear el modelo se ha escogido los siguientes parámetros que se cree que pueden dar un buen resultado:

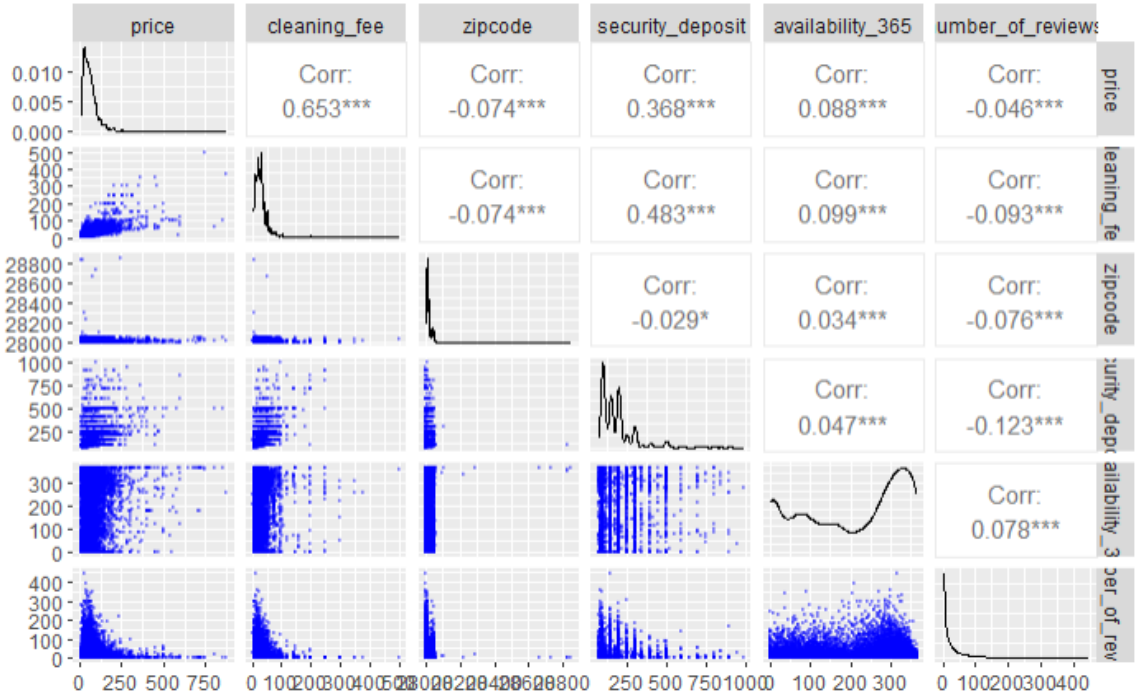
- Price
- Cleaning_fee
- zipcode
- security_deposit
- availability_365
- number_of_reviews

Las características de estos parámetros son:

```
     price        security_deposit  cleaning_fee    availability_365 number_of_reviews review_scores_value
Min.   :  9.00   Min.   : 70.0    Min.   :  4.00   Min.   :  0.0    Min.   :  0.00    Min.   : 2.000
1st Qu.: 31.00   1st Qu.:100.0    1st Qu.: 15.00   1st Qu.: 85.0    1st Qu.:  1.00    1st Qu.: 9.000
Median : 53.00   Median :150.0    Median : 25.00   Median :249.0    Median :  8.00    Median : 9.000
Mean   : 67.41   Mean   :183.5    Mean   : 29.63   Mean   :206.7    Mean   : 23.42    Mean   : 9.212
3rd Qu.: 80.00   3rd Qu.:200.0    3rd Qu.: 35.00   3rd Qu.:321.0    3rd Qu.: 28.00    3rd Qu.:10.000
Max.   :875.00   Max.   :990.0    Max.   :500.00   Max.   :365.0    Max.   :446.00    Max.   :10.000
NA's   :8        NA's   :7391     NA's   :5281                                        NA's   :2795
```

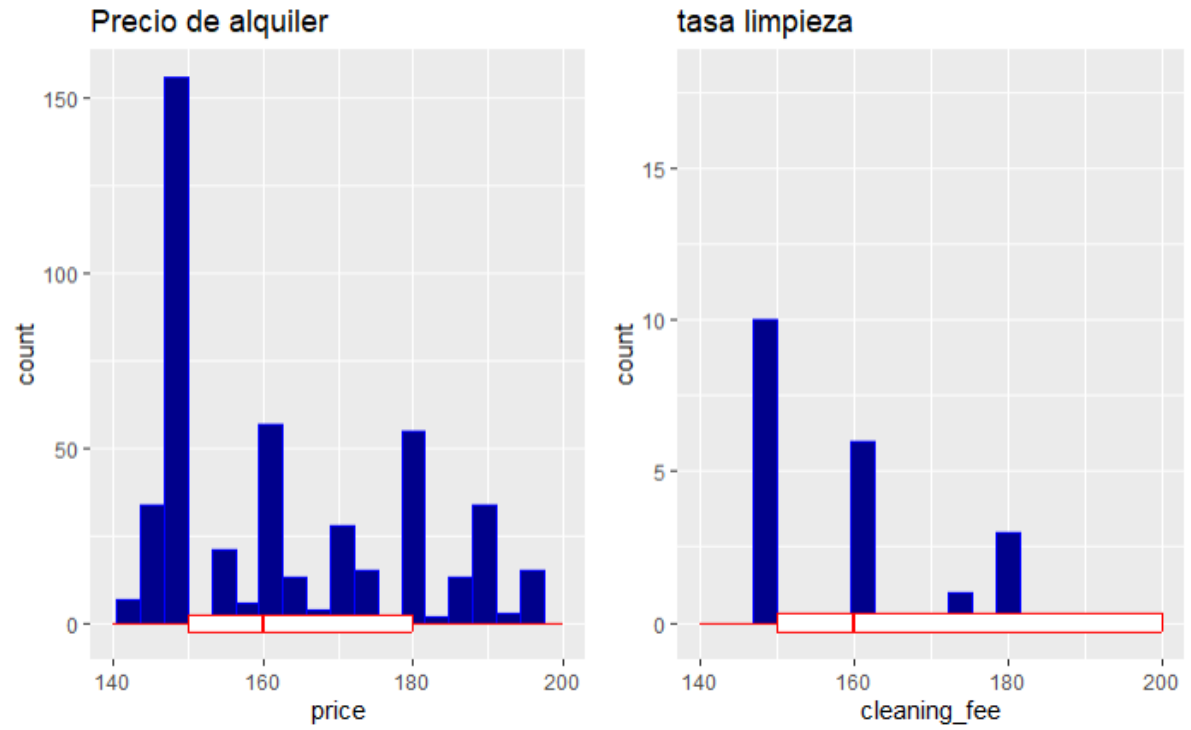
```
   zipcode
Min.   :28001
1st Qu.:28005
Median :28012
Mean   :28016
3rd Qu.:28017
Max.   :28850
```

Vamos a ver cómo se correlacionan cada parámetro con el precio:

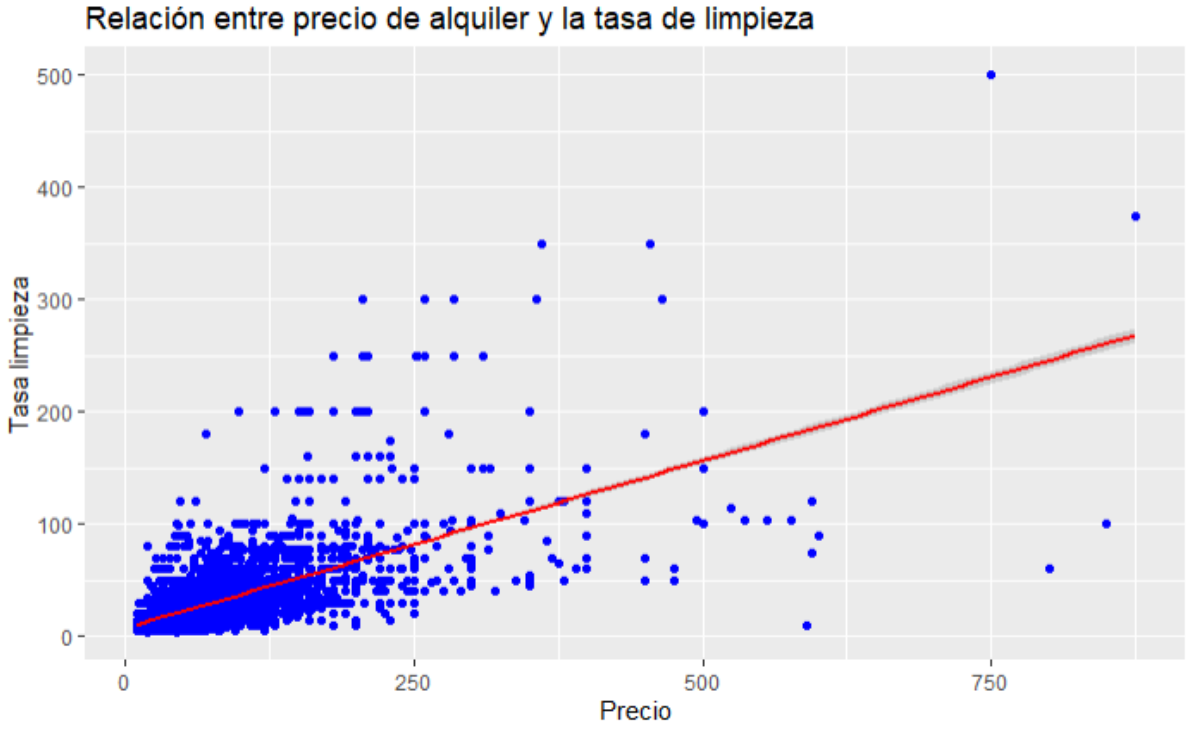


Se puede observar que el “cleaning_fee” es el parámetro de mayor correlación con el precio y se puede observar además por el dibujo que va a ser una correlación positiva, es decir a mayor precio mayor tasa de limpieza.

Respecto a estos dos parámetros también podemos ver el conteo de los alquileres en función del precio y del “cleaning_fee”:

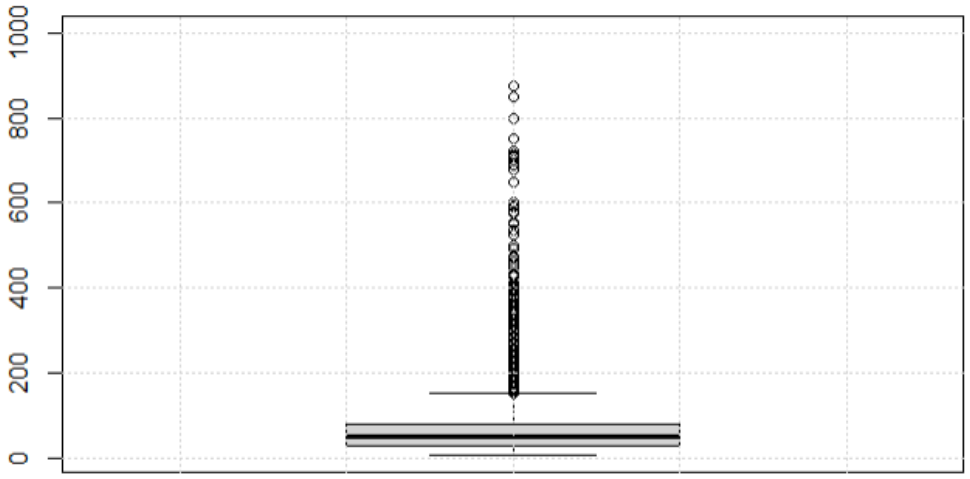


Por tanto para ver esta correlación en más detalle se va a analizar y visualizar un modelo de regresión lineal entre estos dos parámetros, precio y tasa de limpieza.

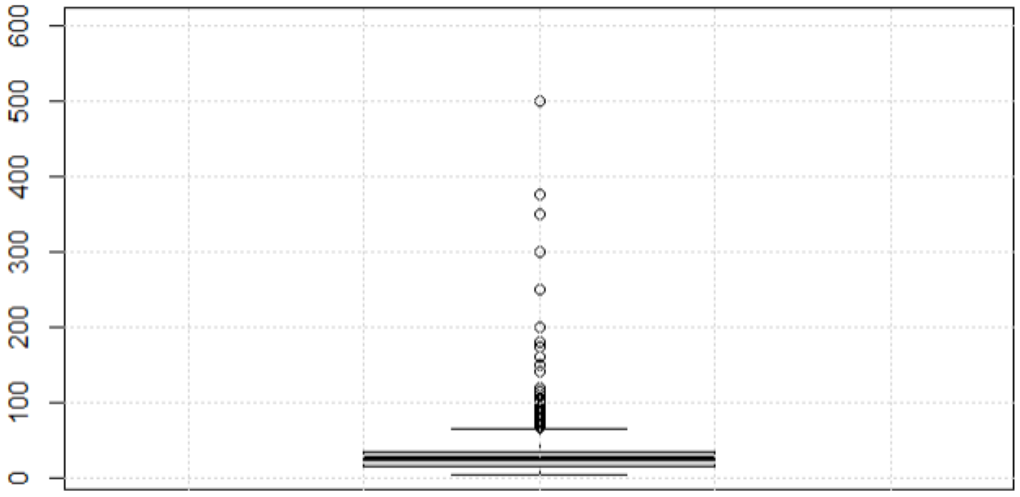


Este modelo tiene un R cuadrático medio de 0.4264. Pero como vemos el gráfico está muy concentrado en valores por debajo de 250 euros indicándonos que seguramente haya outliers. Por tanto a continuación voy a hacer un boxplot del precio y de la tasa de limpieza:

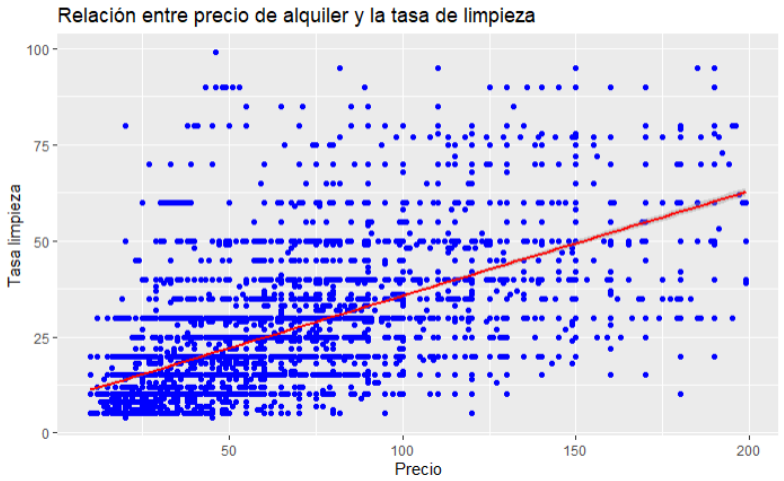
- Boxplot del precio de alquiler:



- Boxplot de la tasa de limpieza

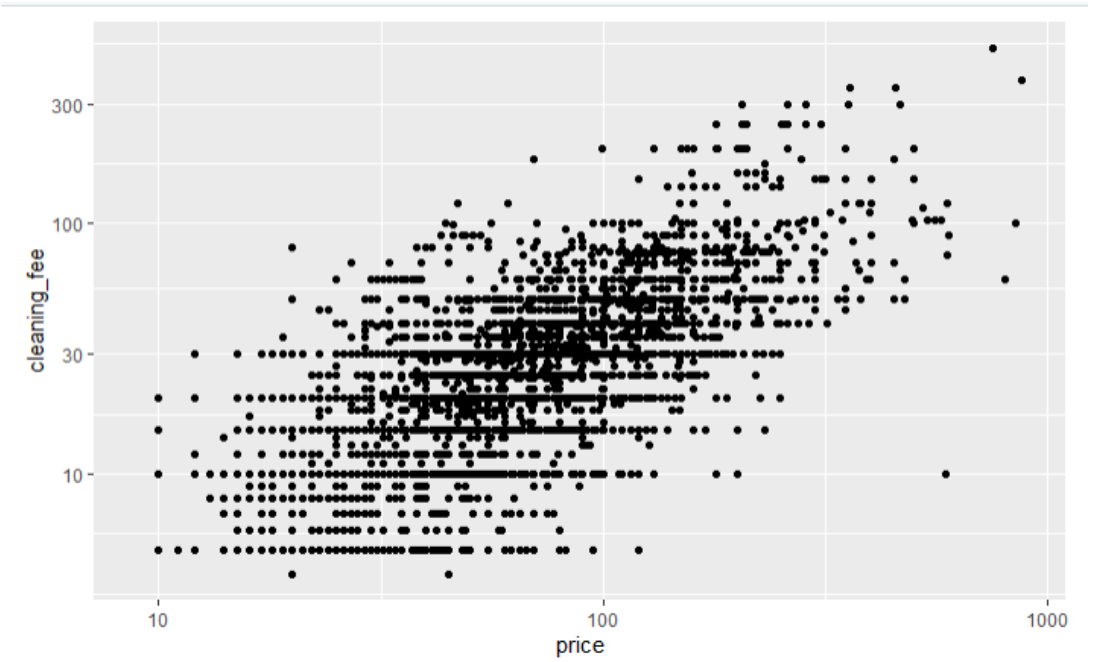


Por tanto podemos ver que hay bastantes outliers por encima de 200€ en el precio y por encima de 100€ en la tasa de limpieza. Así que se va a filtrar el precio y la tasa por valores inferiores a estos y el modelo nos quedaría:

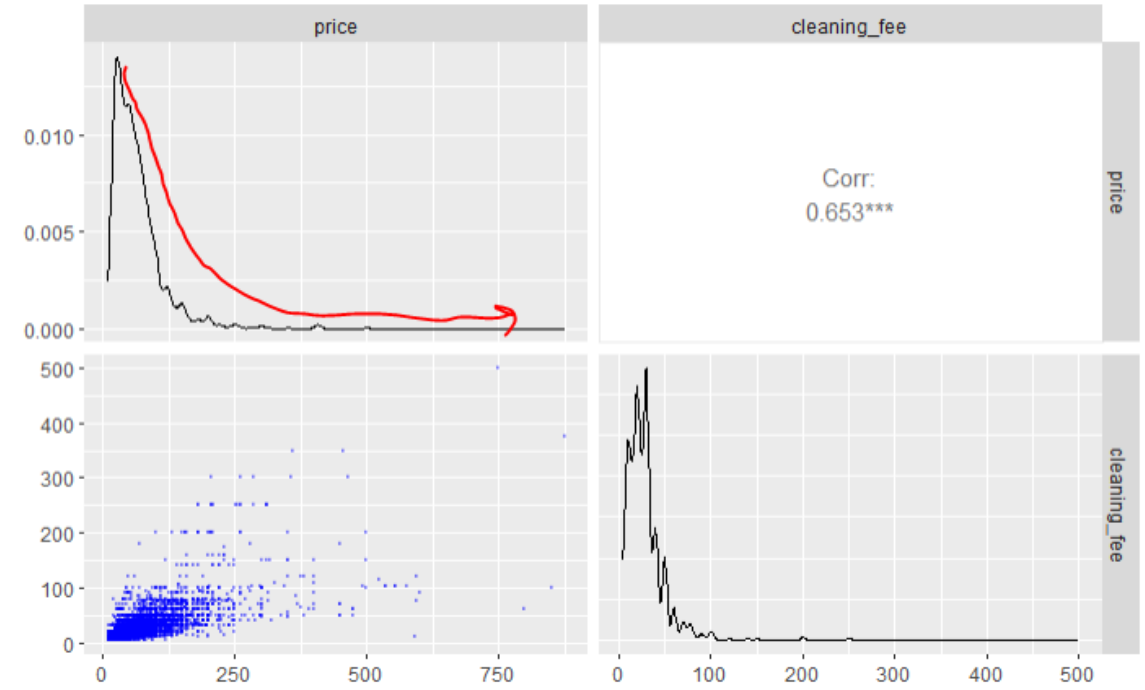


Con este filtro tendríamos 70 filas que cumplen con el precio matoy a 200€ y la tasa de limpieza mayor a 100€ de las 12.872 filas del dataframe.

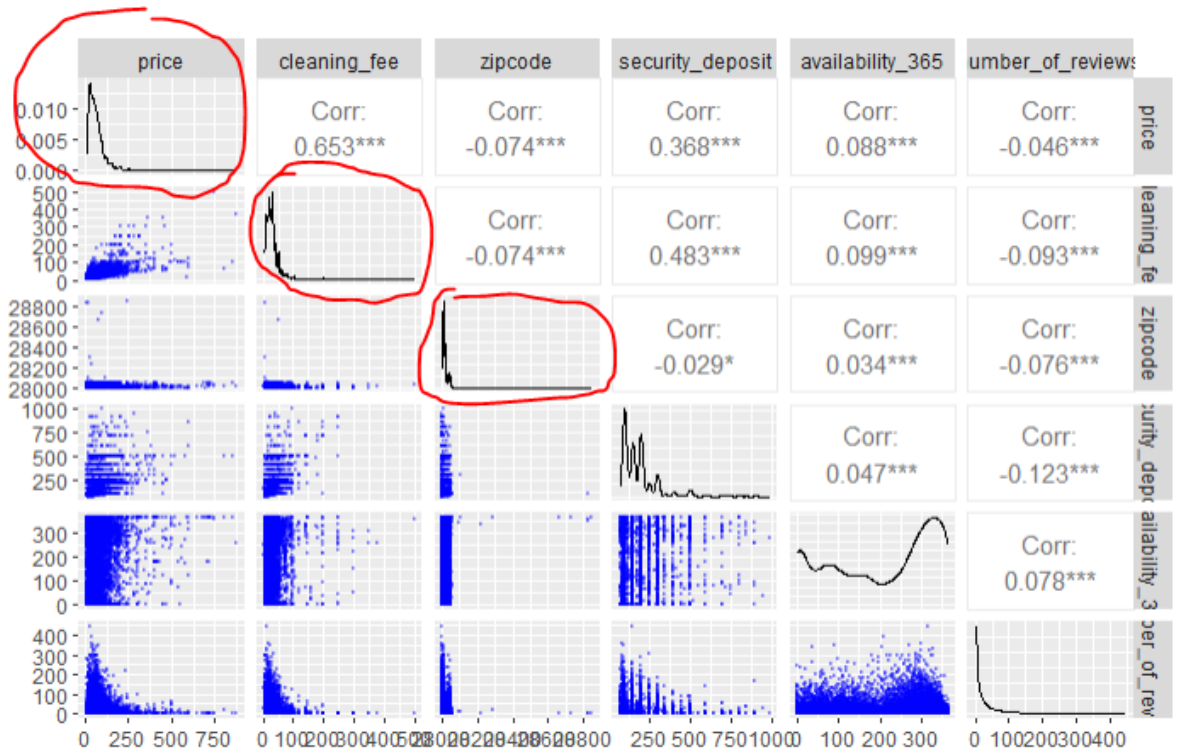
También sin necesidad de filtrar podemos ver como si pasamos los datos a una escala logarítmica se ven a distribuir mejor los datos.



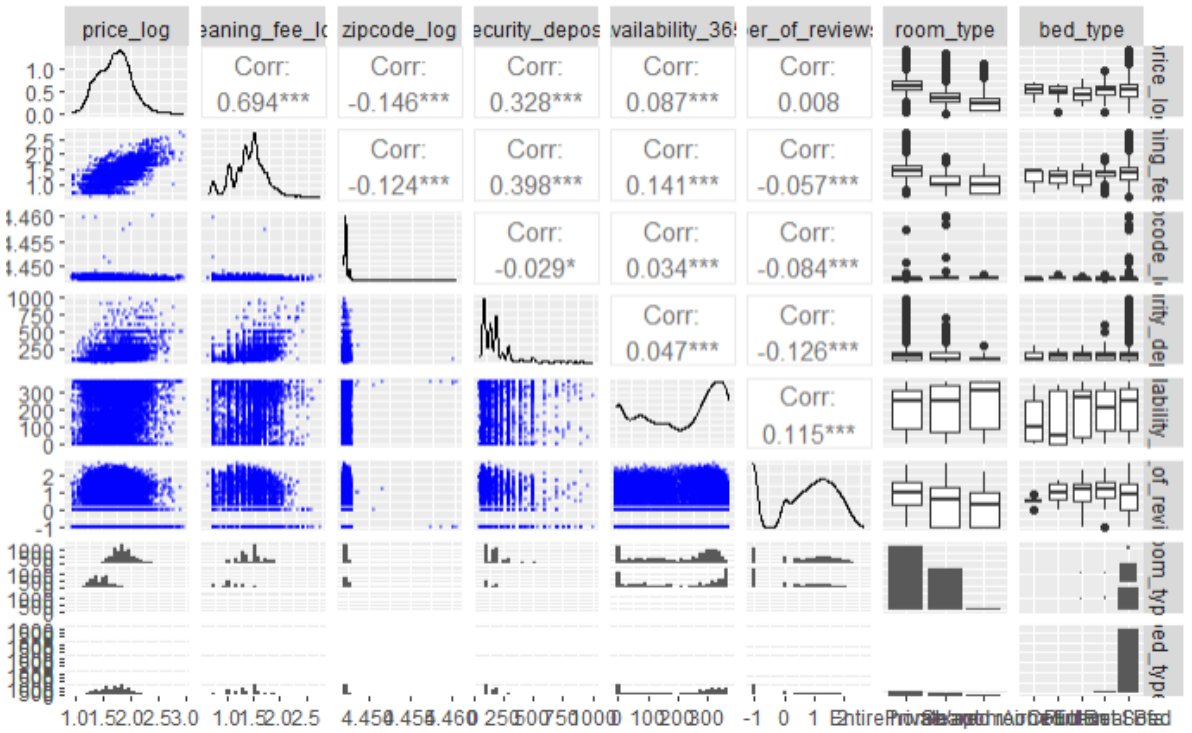
Esto viene bien cuando vemos que la curva del parámetro se acumula bastante al principio y después baja y se alarga como en la siguiente figura:



Por tanto si volvemos a la figura del principio donde tenemos todos lo parámetros podemos ver como el precio, el cleaning_fee y el zipcode tienen esta forma:



Por esto voy a crear tres nuevas columnas en el data frame para poder pasar estos parámetros a escala logarítmica y de nuevo sacando las figuras vemos conseguimos aumentar esta correlación viéndolo en las gráficas mejor:



Después de esto y haciendo varias modificaciones en Rstudio para intentar obtener un modelo con un R cuadrático medio bueno, hemos conseguido de pasar del que teníamos

con solo la correlación entre el precio y la tasa de limpieza que es el que veíamos que tenía la correlación más alta y R cuadrático medio de 0.4264 a un R cuadrático medio de **0.6012.**

Esto lo hemos conseguido con la conversión logarítmica y probando los parámetros que suben el R cuadrático medio. Quedando finalmente una Regresión lineal con un R cuadrático de 0.6012 que va a predecir el precio en función de las siguientes características:

- cleaning_fee
- number_of_revies
- zipcode
- room_type
- bed_type

A continuación vamos a dividir el data frame en training y testing para evaluar la calidad de nuestro modelo.

Utilizaremos 9.010 filas para entrenar el modelo y tendrán las siguientes características:

```
  zipcode_log    cleaning_fee_log   price_log     number_of_reviews_log
Min.   :4.447   Min.   :0.6128   Min.   :0.959   Min.   :-1.00000
1st Qu.:4.447   1st Qu.:1.1790   1st Qu.:1.479   1st Qu.: 0.04139
Median :4.447   Median :1.3997   Median :1.725   Median : 0.90848
Mean   :4.447   Mean   :1.3716   Mean   :1.722   Mean   : 0.67700
3rd Qu.:4.447   3rd Qu.:1.5453   3rd Qu.:1.904   3rd Qu.: 1.46389
Max.   :4.460   Max.   :2.5442   Max.   :2.942   Max.   : 2.59006
                NA's   :1580     NA's   :3
          room_type             bed_type
Entire home/apt:5378   Airbed       :   5
Private room   :3503   Couch        :  11
Shared room    : 129   Futon        :  21
                       Pull-out Sofa: 168
                       Real Bed     :8805
```

Para test utilizaremos 3.862 filas aleatorias del dataframe con la siguientes características:

```
  zipcode_log    cleaning_fee_log   price_log     number_of_reviews_log
Min.   :4.447   Min.   :0.6128   Min.   :1.004   Min.   :-1.00000
1st Qu.:4.447   1st Qu.:1.1790   1st Qu.:1.520   1st Qu.: 0.04139
Median :4.447   Median :1.3997   Median :1.717   Median : 0.90848
Mean   :4.447   Mean   :1.3716   Mean   :1.720   Mean   : 0.67700
3rd Qu.:4.447   3rd Qu.:1.5453   3rd Qu.:1.904   3rd Qu.: 1.46389
Max.   :4.460   Max.   :2.5442   Max.   :2.859   Max.   : 2.59006
                NA's   :1580     NA's   :5
          room_type             bed_type
Entire home/apt:2333   Airbed       :   0
Private room   :1477   Couch        :   4
Shared room    :  52   Futon        :  14
                       Pull-out Sofa:  64
                       Real Bed     :3780
```

Si vemos la calidad de ambos tienen un R cuadrático medio parecido así que nos sirven:

La calidad del modelo medida en Training:

```{r}
df_bnb.train$pred_log <- predict(final_model, df_bnb.train)
caret::postResample(pred=df_bnb.train$pred_log, obs=df_bnb.train$price_log)
```

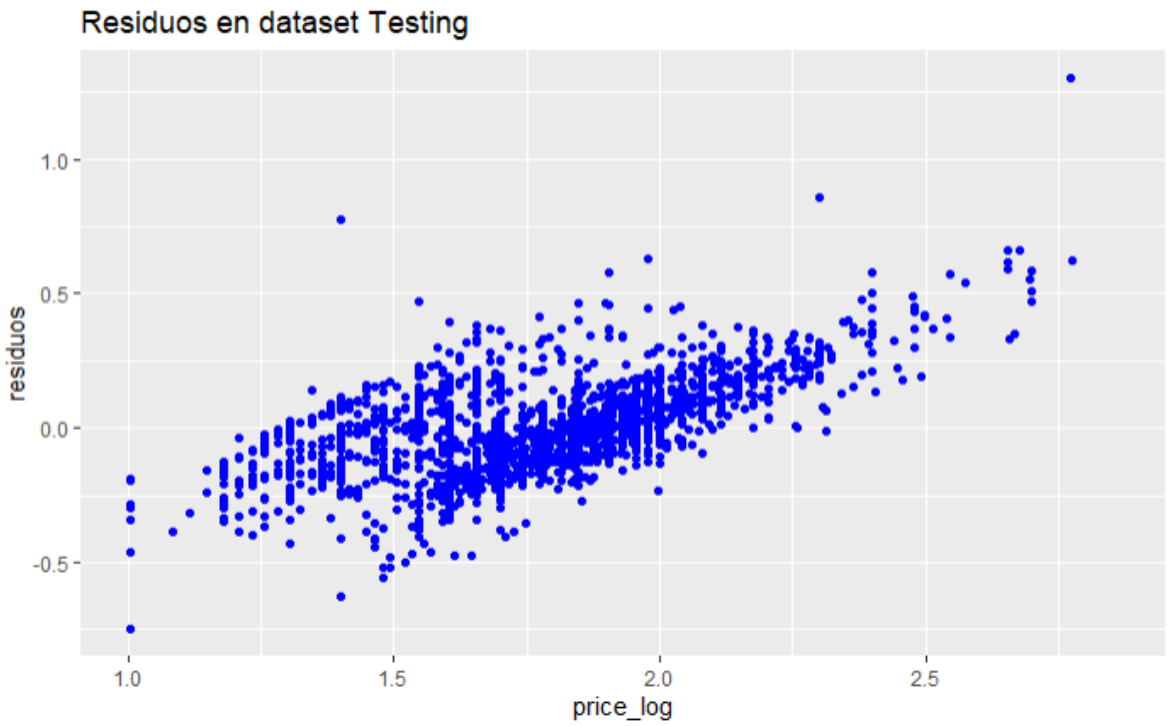
```
      RMSE   Rsquared        MAE
        NA  0.6048135         NA
```

La calidad del modelo en testing (real):

```{r}
df_bnb.test$pred_log <- predict(final_model, df_bnb.test)
caret::postResample(pred=df_bnb.test$pred_log, obs=df_bnb.test$price_log)
```

```
      RMSE   Rsquared        MAE
        NA  0.5926244         NA
```

Por último vamos a ver los residuos de testing:



6. Conclusiones y “lessons learned”

Comenzamos el trabajo partiendo de las suposiciones iniciales. Lo que nos ayudó a seleccionar los campos que creíamos necesarios para verificar nuestras suposiciones. La limpieza de datos fue una de las partes más importantes del proceso, ya que nos íbamos dando cuenta que no nos cuadraban muchos de los datos a la hora de seguir con las siguientes partes del proyecto, como las visualizaciones de Tableau o los gráficos en R. Por

lo que tuvimos que volver una y otra vez a los pandas para seguir puliendo los datos y obtener los mejores resultados posibles.

Como comentábamos en la introducción, partimos de unas suposiciones iniciales, como por ejemplo, que según el barrio de Madrid que seleccionamos variaría el precio en comparación con otros barrios, ya fuera por encontrarse en el extrarradio de la ciudad o por los servicios cercanos al apartamento. Trás evaluar la relación entre los campos de localización de barrio y el precio, vimos que no siempre se cumplía esta premisa.

Otra de las hipótesis que nos planteamos es que si un anuncio no tenía fotos el número de reservas sería menor. No pudimos verificar o desmentir dicho punto puesto que todos los campos venían con el campo de foto informado.

Por otro lado, también nos planteamos que el número de reseñas influía en el precio del alquiler. Como hemos visto en puntos anteriores, en el modelo de regresión lineal, comprobamos que a mayor número de reseñas tiene el precio más bajo, por lo que parece que cuanto más barato, más se alquila y más opiniones tiene.

Para los desarrollos hemos utilizado las métricas que necesitábamos para las suposiciones iniciales, excepto las utilizadas para el campo de las imágenes, ya que, como hemos dicho, rápidamente descartamos el campo al comprobar que no nos iba a aportar utilidad.
Durante la limpieza y realización del proyecto hemos ido descartando muchas métricas. Partíamos de unas 91 y redujimos a 31 en las primeras limpiezas. Finalmente hemos utilizado eficazmente 5 métricas de las contenidas en el Dataset.

Teniendo en cuenta todo el proceso del proyecto y lo aprendido durante los módulos del curso, nos hemos dado cuenta que partíamos de ideas preconcebidas, las cuales nos sesgaban los datos del principio y nos hemos dado cuenta que deberíamos haber comenzado por el análisis de datos tal y cómo se nos mostraban y de ahí plantear las necesidades de examinar los datos para crear cruces con fiabilidad.
Los módulos trabajados durante el curso nos han ayudado a poder hacer preguntas que con análisis de datos podamos resolver y comprender la relación entre ellos. Ha sido por ello que finalmente hemos conseguido conclusiones fiables basadas en los resultados, pero tras muchas comprobaciones, cambios, puestas en común, ideas variadas.