Trabajo Practico N°2 Big Data (UNT) 2024

Grupo I:

Pilar Valdez

José López

Parte I: Limpieza de la Base

Detectamos que algunos datos estaban desplazados a nivel de Fila, y que se podían acomodar para no perderlos. Como no pudimos encontrar una solución mediante código de Python, acomodamos esos datos usando Stata y pasando a una planilla Excel, desde la cual cargamos los datos en Jupyter.

Detectamos 11 filas duplicadas. Las cuales fueron eliminadas.

Observamos que para 10 de las 16 variables habían exactamente 285 valores faltantes. Estas eran: host_id, neighbourhood, neighbourhood_group, latitude, longitude, room_type, minimum_nights, number_of_reviews, calculated_host_listings_count y availability_365. Usamos el comando preserve y probamos eliminando las observaciones para las cuales neighbourhood tenía missing values. Observamos que al eliminar esas 285 variables para neighbourhood, se eliminaron los missing values para las otras 9 variables y se redujeron en la misma cantidad los datos faltantes en 4 de las 6 variables restantes. Esto implicaba que para esas 285 observaciones teníamos sólo dos variables de información, por lo que imputar esas observaciones no hubiera sido posible.

Observamos que la variable id pasó de tener 161 observaciones antes de eliminar las 286 a tener 160 observaciones luego de haberlo hecho. Si bien al momento de hacer la limpieza nos percatamos de que 160 observaciones se habían desplazado una fila hacia la izquierda en la tabla, no sucedía lo mismo con los id, por lo que, como además contamos con una segunda variable de identificación del host, host_id, y consideramos no es relevante para el análisis, es que decidimos eliminarla del data frame.

Por lo que ahora teníamos sólo 6 variables que presentaban datos faltantes. A las variables name y host_name, con 26 y 22 datos faltantes respectivamente, decidimos eliminarlas porque consideramos que no añadirían valor al análisis de datos ni a la predicción de ningún efecto que pudiera interesarnos estimar. Cabe aclarar que la eliminación de estas variables, al igual que la de host_id, fue posterior a la de las observaciones duplicadas.

Obsevamos que cuando reviews per month está vacío entonces Last reviews esta vacío es decir que no tiene valores porque no tuvieron nunca reseñas y que number of reviews siempre es cero pues no hubo reseñas. Es por ello que no son valores perdidos, sino que se tratan de campos que es correcto no tengan valor.

A continuación reemplazamos los vacíos de reviews per month por valor de cero.

A los 15 datos faltantes en la variable Price, les realizamos una imputación por media condicional al tipo de habitación (room_type) y al vecindario (neighbourhood). En todos los casos teníamos al menos 15 y 1700 precios para los pares de tipos de habitación y vecindarios que correspondían a los missing values de price.

Los anfitriones en el mismo vecindario suelen ajustar sus precios en respuesta a lo que otros están cobrando, creando un mercado competitivo que limita las variaciones de precios para habitaciones similares. Habitaciones con características similares en el mismo vecindario atraerán a un público similar y, por lo tanto, deberían tener precios comparables. Es decir que bajo el supuesto de mercado competitivo hacer este supuesto no es descabellado cuando tenemos muchos oferentes y muchos demandantes.

Elegimos la media condicional a neighbourhood en lugar de neighbourhood_group, porque se trata de una instancia de datos mas desagregada.

Transformamos las variables 'neighbourhood_group' y 'room_type' a variables numéricas. Para ello tuvimos en cuenta que los valores de room_type fueran en orden creciente de acuerdo a la calidad de la habitación, quedando:

'Entire home/apt': 1,

'Private room': 2,

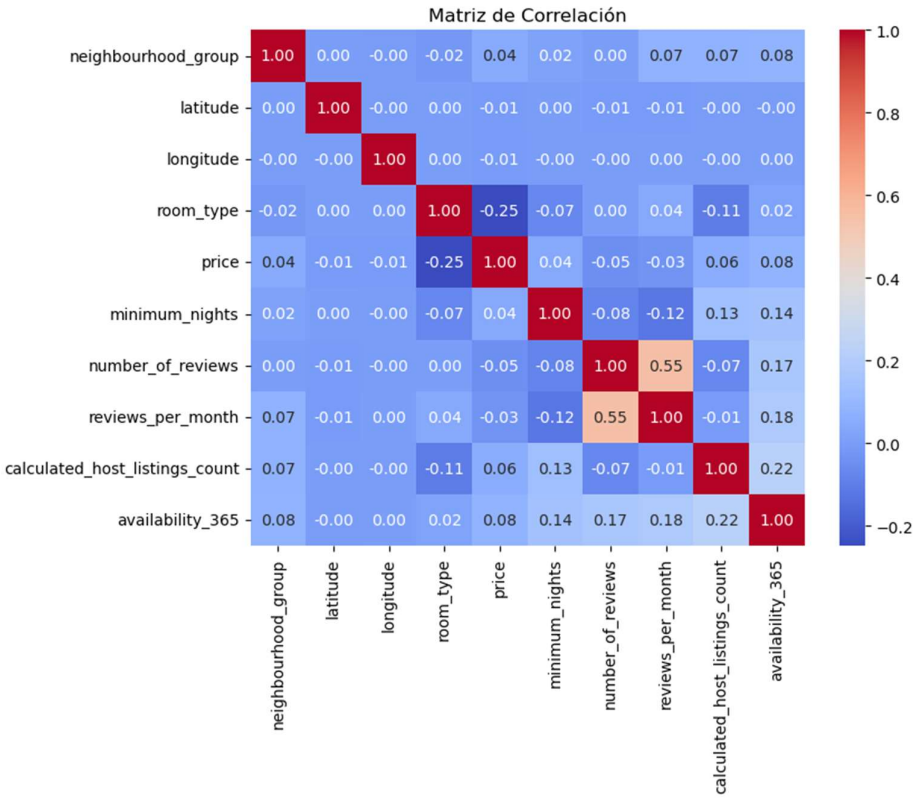
'Shared room': 3

Y la variable **'neighbourhood_group'** fue transformada a numérica teniendo en cuenta la cantidad de ofertas en cada ciudad ordenadas de forma creciente**:**

```
[231]: #Por otro lado transformamos la variable neighbourhood_group del siguiente modo:
mapeo_neighbourhood_group = {
    'Manhattan': 1,
    'Brooklyn': 2,
    'Queens': 3,
    'Bronx': 4,
    'Staten Island': 5
}
df['neighbourhood_group'] = df['neighbourhood_group'].map(mapeo_neighbourhood_group)
frecuencias = df['neighbourhood_group'].value_counts()
print(frecuencias)
```

```
neighbourhood_group
1    21635
2    20065
3     5627
4     1090
5      372
Name: count, dtype: int64
```

Parte II: Gráficos y visualizaciones



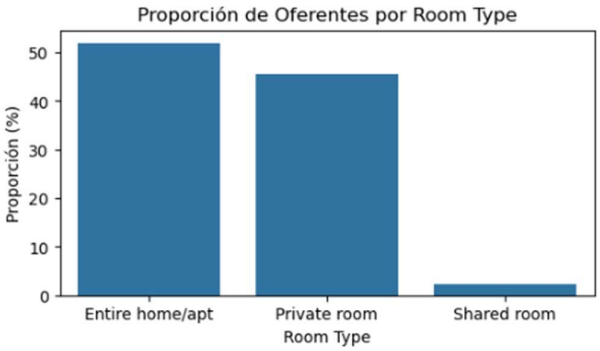
Existe una correlación negativa igual a -0.25 entre el precio y el tipo de habitación, dada la manera de numerar el tipo de habitación, lo anterior significa que el precio cae al pasar de una casa en alquiler a una habitación privada a una habitación compartida.

La variable “calculated_host_listings_count” tiene correlaciones positivas con la disponibilidad (0.22) y con el mínimo de noches (0.13). lo cual significa que cuanto mas

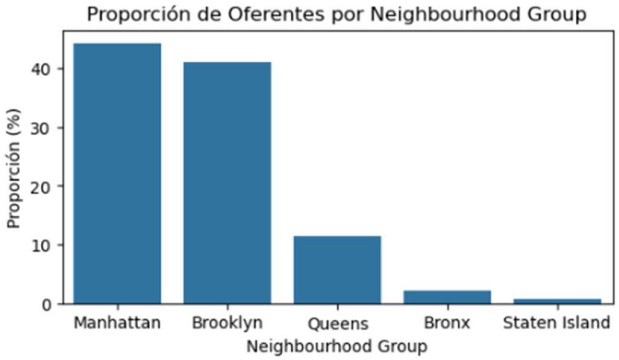
alojamientos ofrece un mismo anfitrión, generalmente esta disponible mayor tiempo y además el mínimo de noches es mayor.

La variable disponibilidad “availability_365” es la variable con más correlaciones positivas, se correlaciona positivamente con las reseñas, con el mínimo de noches, con el precio y con la “calculated_host_listings_count” como describimos anteriormente.

Respondan las siguientes preguntas: ¿Cuál es la proporción de oferentes por “Neighbourhood group”? ¿Y por tipo de habitación? Además, realicen gráficos para mostrar estas composiciones y comenten los resultados.



La mayoría ofrecen casa completa (52%), seguidos de habitación privada (45,7%) y habitación compartida (2.34%).



Manhatan y Brooklin aglutinan el 85% de las ofertas.

Realicen un histograma de los precios de los alojamientos. Comenten el gráfico obtenido. Además, respondan las siguientes preguntas: ¿cuál es el precio mínimo, máximo y promedio? ¿Cuál es la media de precio por “Neighbourhood group” y por tipo de habitación?

Promedio: 152.78111254586076

Máximo: 10000.0

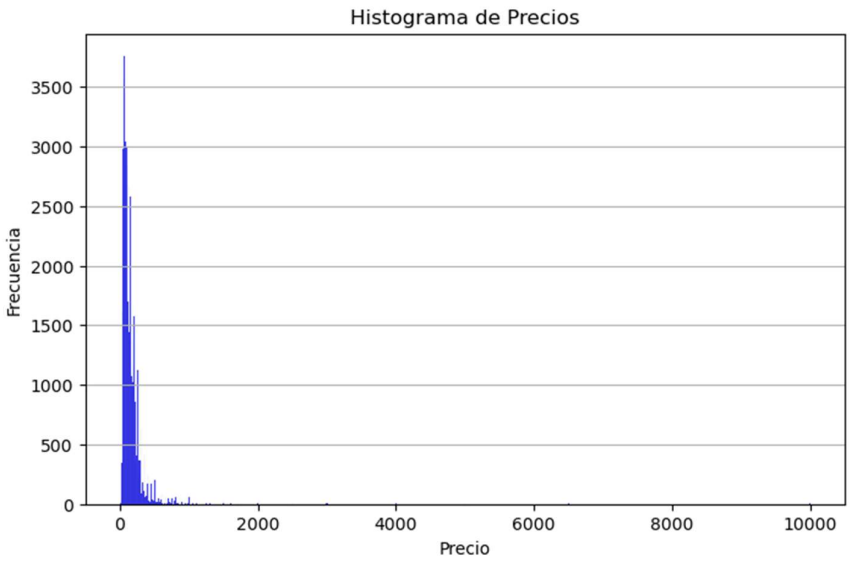
Mínimo: 0.0

Promedio: 128.6111258977609

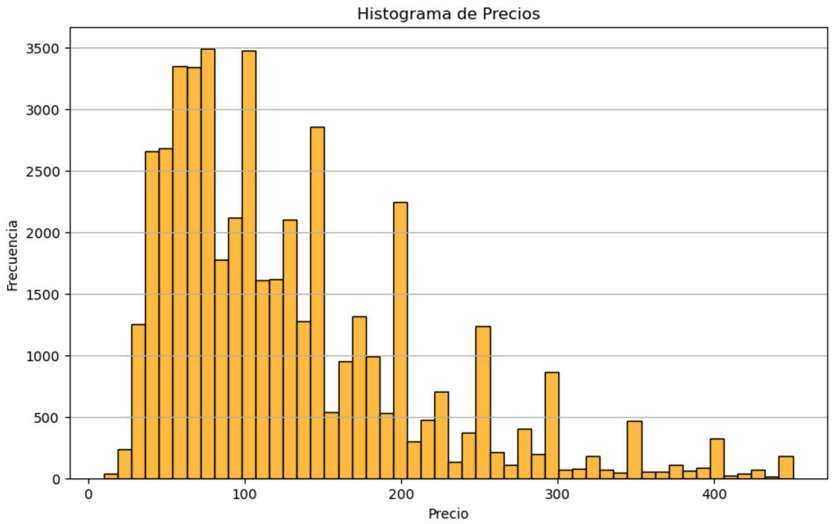
Máximo: 450.0

Mínimo: 10.0

Aplicamos un histograma directamente de los datos, pero la presencia de valores extremos desvirtuaba el análisis.

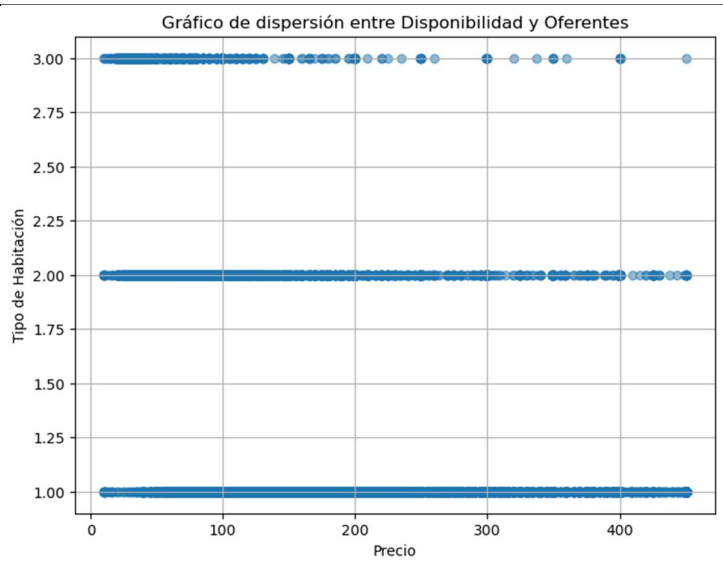
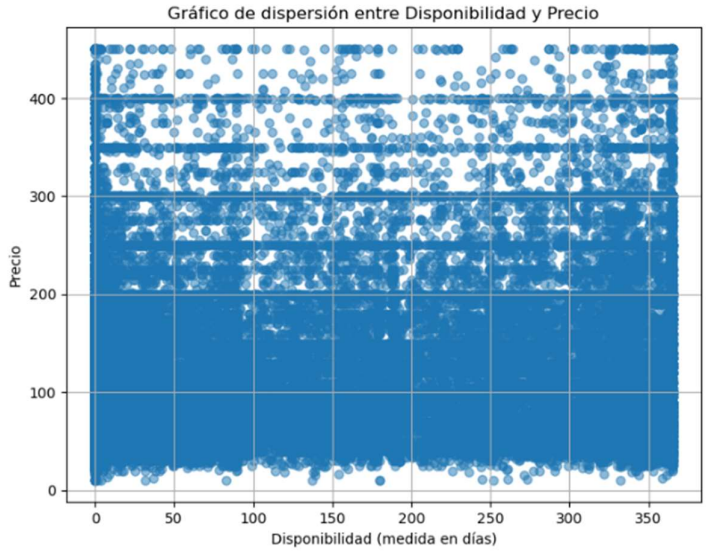


Luego de quitar los valores extremos al momento de graficar obtenemos:



	room_type	neighbourhood_group	price
0	1	1	120.284182
1	1	2	156.937420
2	1	3	197.868250
3	1	4	136.901067
4	1	5	125.071006
5	2	1	60.718547
6	2	2	71.085115
7	2	3	102.083928
8	2	4	66.834440
9	2	5	62.292553
10	3	1	47.254237
11	3	2	49.127139
12	3	3	82.350211
13	3	4	48.127072
14	3	5	57.444444

Realicen dos scatter plots con dos variables de interés en cada uno. Comenten

	Se Observa que al pasar de casa a habitación privada a habitación compartida el precio cae.
	No se Observa una relación directa entre disponibilidad y precio, si bien tienen coeficiente de correlación positivo

Análisis de Componentes Principales

Porcentaje de varianza explicada por los dos componentes: 51.86%, por lo que estas dos nuevas variables resumen una gran parte de la información original.

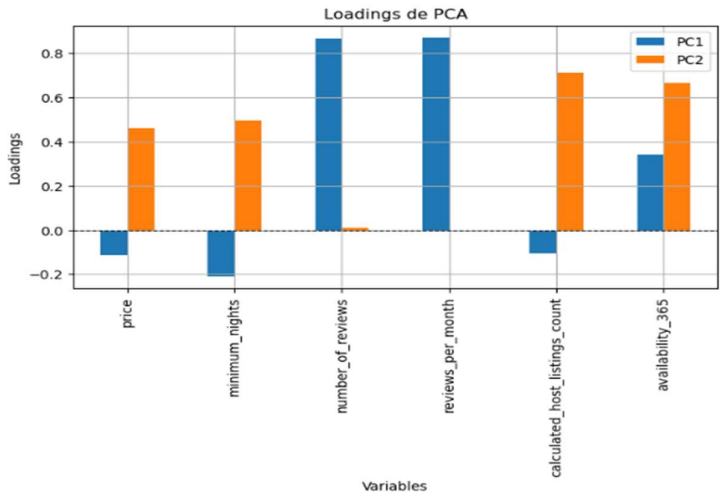
```
                                     PC1       PC2
price                          -0.112393  0.462762
minimum_nights                 -0.209615  0.495876
number_of_reviews               0.868088  0.010771
reviews_per_month               0.871635 -0.005453
calculated_host_listings_count -0.107669  0.712339
availability_365                0.341542  0.667720
<Figure size 800x400 with 0 Axes>
```

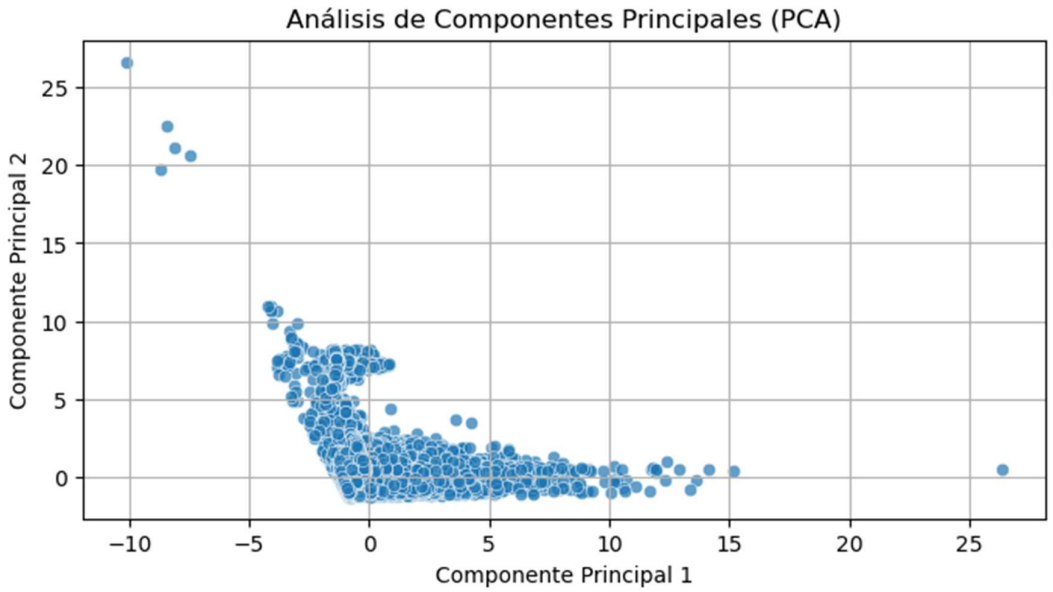
PC1:

- number_of_reviews y reviews_per_month: Tienen altas cargas positivas en PC1.
- price: Tiene una carga negativa en PC1.
- Availability_365 tiene también carga positiva en PC1.
- Interpretación general: PC1 podría representar una dimensión relacionada con la popularidad o actividad de un alquiler.

PC2:

- calculated_host_listings_count y availability_365: Tienen altas cargas positivas en PC2.
- minimum_nights: Tiene una carga positiva en PC2.
- Interpretación general: PC2 podría representar una dimensión relacionada con la oferta del anfitrión o la disponibilidad del alquiler.





Predicción de los Precios de los Alojamientos

La Regresión se realiza usando como variable dependiente Precio y como regresores

number_of_reviews, neighbourhood, latitude, longitude, room_type, minimum_nights, reviews_per_month, host_listings, availability.

Coeficiente de determinación: 0.3473102993985592

Intercepto: 236.60569882899904

Pendiente: [-7.48177458e-02 6.38013568e+00 -4.47676554e-08 5.23326321e-08

-8.45392571e+01 -1.78257377e-01 -1.77965996e+00 2.04517555e-01 + 6.37666696e-02]