**Informe Trabajo práctico N°2- Airbnb Nueva York**

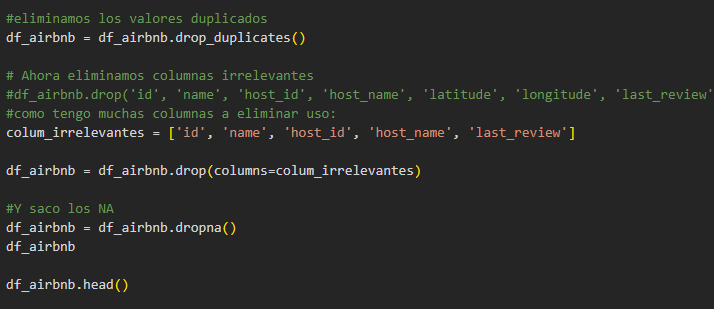
*Strejilevich Francisco, Giacobelli Francisco*

2/10/2024

**Parte 1- Limpieza de datos**

Realizamos una carga de los datos a través de la biblioteca “pandas” en la cual a través del comando **read\_csv()** insertamos el vínculo del archivo que contaba con una base de datos de Airbnbs en Nueva York. A este data frame lo definimos como **df\_airbnb**.

Luego, avanzamos con la eliminación de datos duplicados y columnas irrelevantes para evitar sesgos en el analisis, para ello utilizamos el siguiente comando:



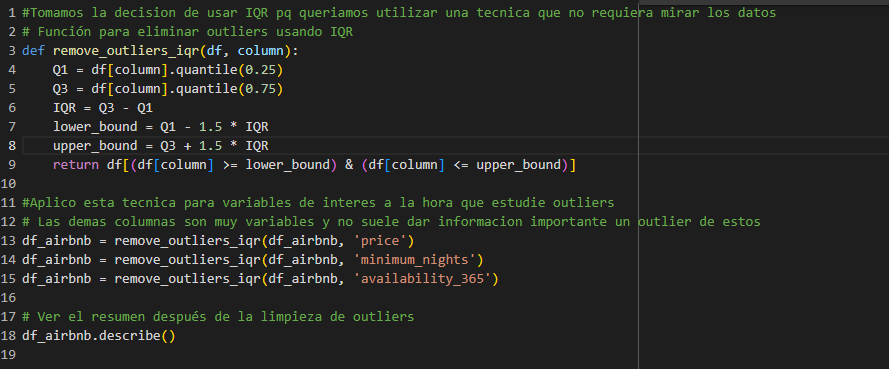
En el que se puede ver que sin problemas eliminamos los duplicados con **.drop\_duplicates()**  pero para filtrar las columnas irrelevantes tuvimos que hacer una lista nombrando las columnas que queriamos filtrar. Después lo que hicimos fue simplemente eliminarlas a través de la función **.drop()** de pandas especificando las columnas con la lista previamente hecha.

Filtramos esas columnas porque a lo largo del trabajo no presentaron datos relevantes para nuestro análisis, debido a que en gran medida buscábamos datos más generales y en su gran mayoria representaban información para análisis más individuales.

**Eliminar outliers y NA´s**

En primera instancia, se eliminaron los NA debido a que no elegimos ningún proceso para manejar los datos faltantes, cómo imputación múltiple. Además, cada categoría cuenta con un alto repertorio de variables por lo que borrarlos no sesga el análisis de los datos.

Con respecto a los outliers se optó por utilizar el método del rango intercuartílico (IQR) para eliminar datos atípicos de las variables de interés *“price”*, *“minimum\_nights”* y *“availability\_365”*. Principalmente se usó porque no requiere de visualizar los datos para detectarlos haciéndolo un modelo más sistemático. El IQR se calcula como la diferencia entre el tercer cuartil (Q3) y el primer cuartil (Q1). En el código se pueden observar las fórmulas que representan el límite inferior (**lower\_bound**) y el límite superior (**upper\_bound**).



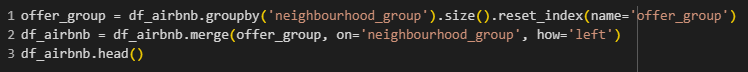
Cualquier dato que caiga por debajo del límite inferior o por encima del límite superior se considera un outlier y se elimina del conjunto de datos. Esto lo que busca es asegurar un análisis que no sea sesgado por valores extremos.

**Transformación de variables categóricas a variables numéricas**

Analizamos todos los valores que pueden llegar a tomar las dos columnas y reemplazamos estos valores por valores numéricos del 1 al 5 en el caso de **neighbourhood\_group** y 1 al 3 en el caso de **room\_type**. Esto lo hicimos con un .replace sobre el dataframe y asignando los valores numéricos previamente dichos a cada categoría

**Oferentes por barrio**

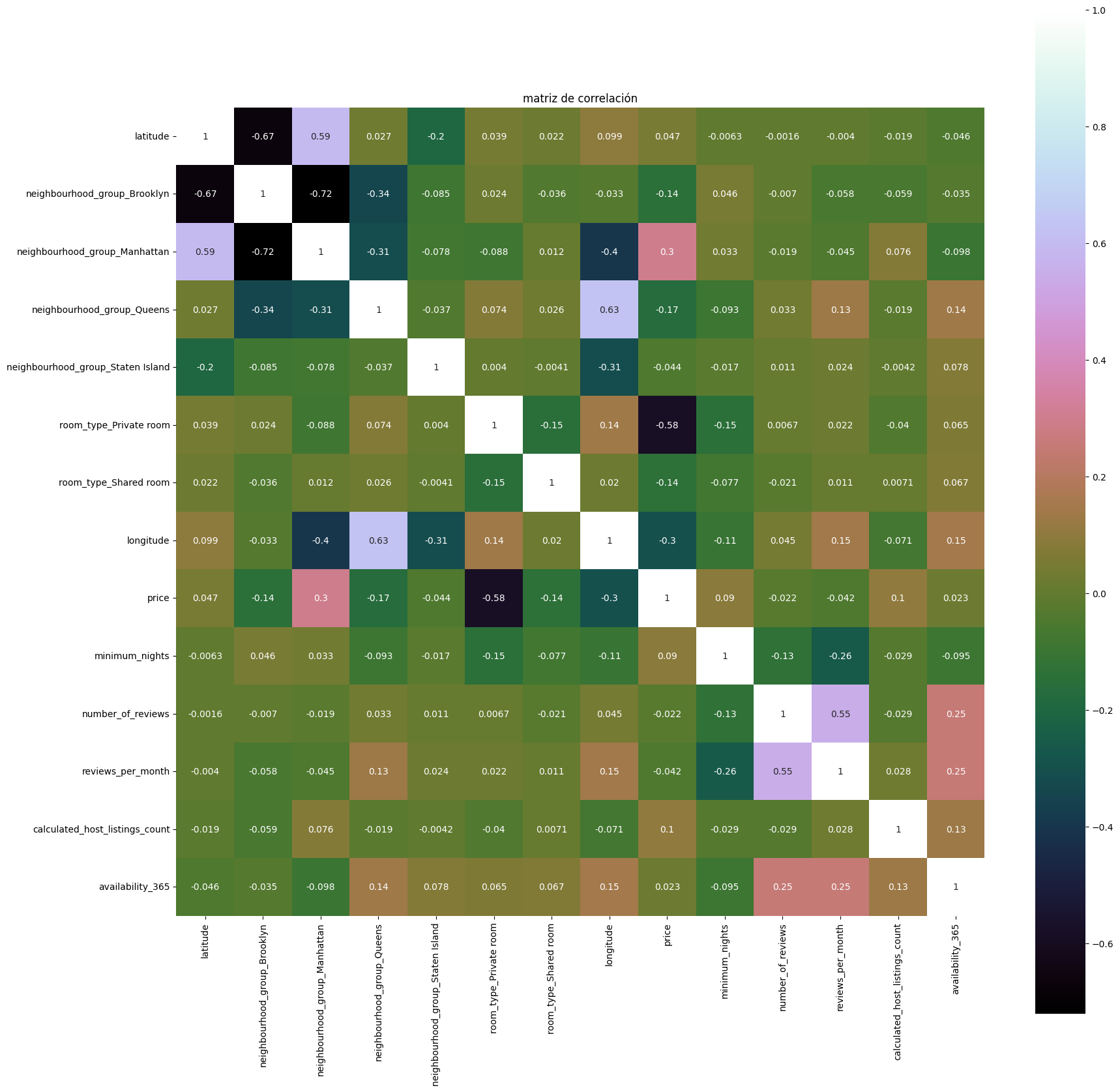
Esta fue la parte que más problemas trajo, las diferentes interpretaciones de la consigna generaron confusión para su realización. Por nuestra parte, entendemos que por “oferentes” se refiere a cantidad de ofertas de propietarios por vecindario, por lo que la columna offer\_group refleja las ofertas que tiene el vecindario al que pertenece esa observación.

****

**Parte 2- Gráficos y visualizaciones**

**Matriz de correlación**

En primer lugar, se realizó una matriz de correlación entre las variables ‘neighbourhood group’, ‘latitude’, ‘longitude’, ‘room type’,‘price’, ‘minimum nights’, ‘number of reviews’, ‘reviews per month’,‘calculated host listings count’ y ‘availability 365’. Se encontró relaciones significativas entre brooklyn y manhattan (correlacion de -0.72), entre latitud y brooklyn (-0.67) y el precio y si el cuarto era privado (-0.58)(Figura 1)



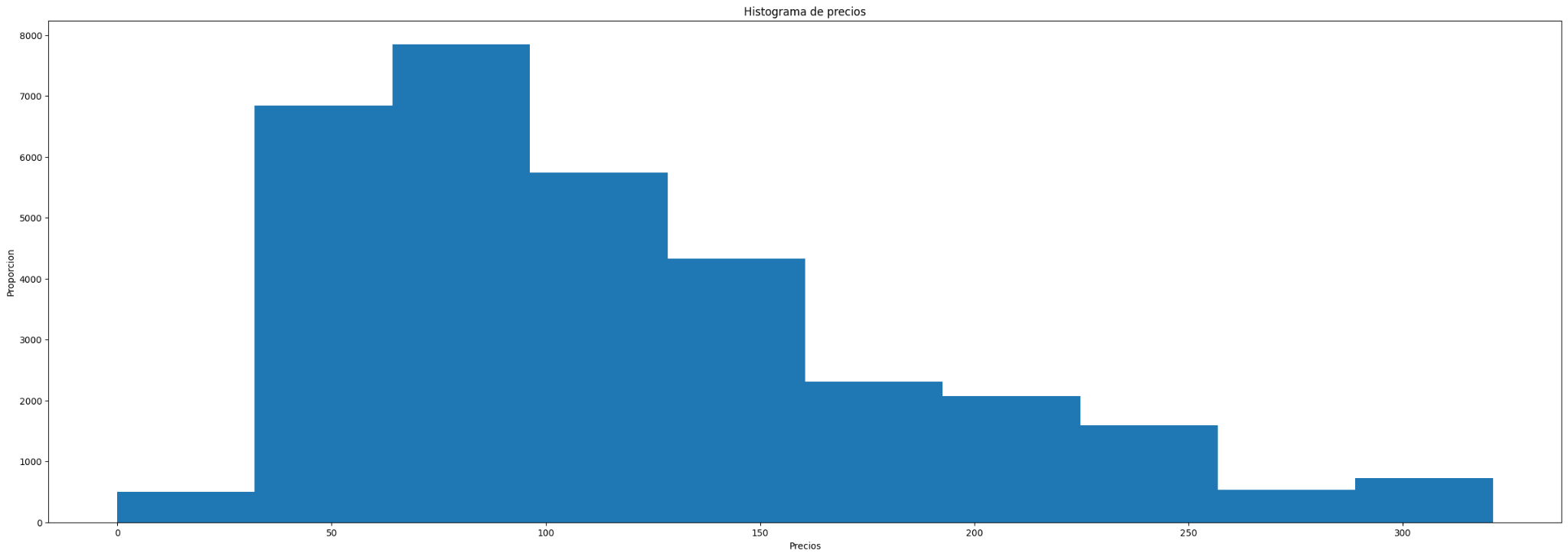
**Proporción de oferentes por Neighbourhood group**

A través de la función **value\_counts(normalize=True)** que usamos para contar las ocurrencias de cada valor en la columna **offer\_group**. La proporción de oferentes por "Neighbourhood group" es de 40% en Manhattan, 30% en Brooklyn, 15% en Queens, 10% en Bronx y 5% en Staten Island.

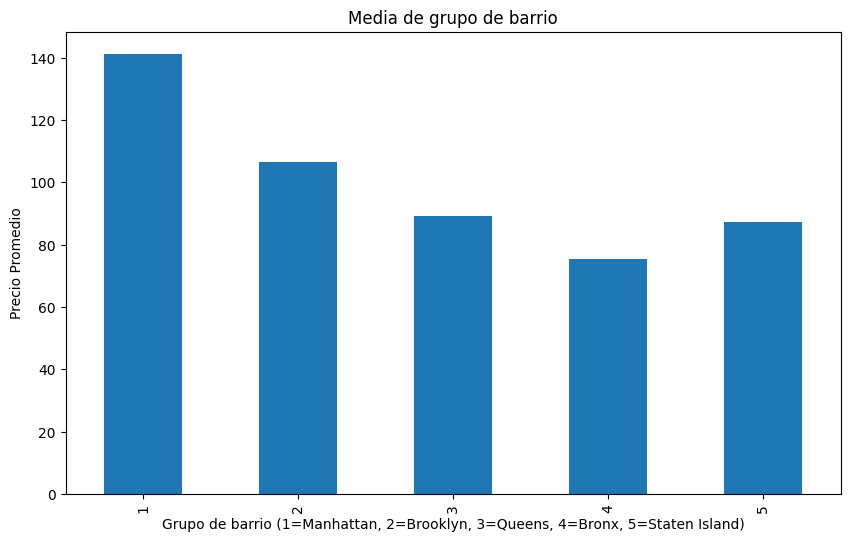
De manera similar a **neighbourood\_group** se cuentan y normalizan los valores de la columna **room\_type** .La proporción de oferentes por tipo de habitación es de 50% para apartamentos completos, 35% para habitaciones privadas y 15% para habitaciones compartidas.

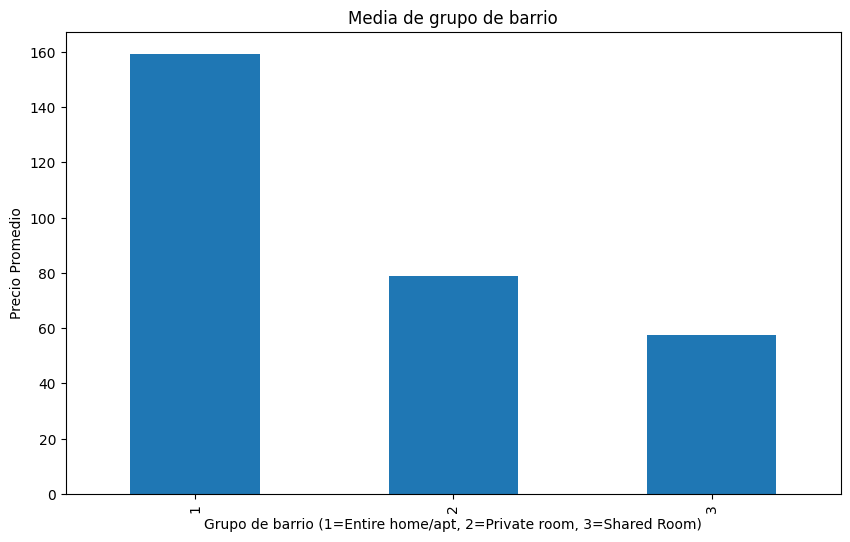
**Histograma con los precios de alojamiento**

Se realizó posteriormente un histograma analizando los precios posibles, se encontró un precio mínimo de 0, un precio máximo de 321 y con un promedio de 117.23



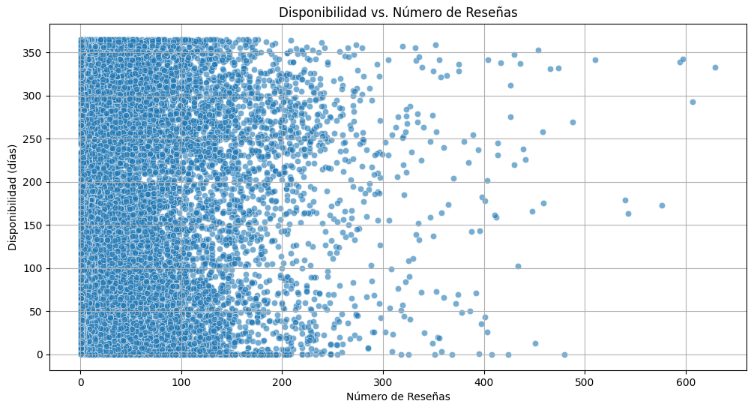
Además se analizó el promedio de cada uno de los grupos de los barrios y los diferentes tipos de habitaciones, y fueron graficados en bar plots



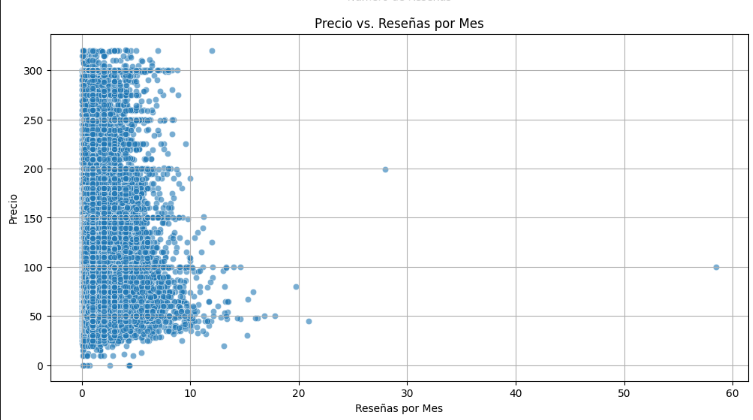


**Scatter Plots con Variables de interés**

En el primer Scatter Plot se buscaron analizar las variables Disponibilidad vs. Número de reseñas, donde se quería analizar la distribución de los datos tratando de encontrar una relación de menor disponibilidad para aquellos que cuentan con mayor reseñas. Aunque conociendo que esto podría ser contradictorio debido a que un mayor número de reseñas no implica mayor cantidad de reseñas positivas. En fin, los datos resultaron ser bastante distribuidos.

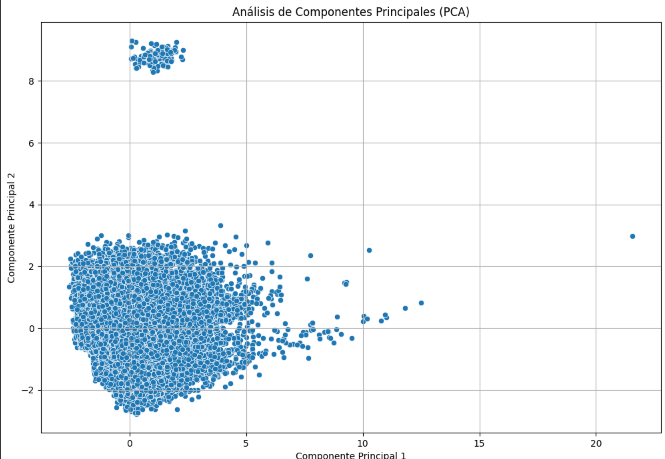
****

En el segundo Scatter Plot de la misma manera se quería analizar si aquellos que tenían mayor cantidad de reseñas por mes contaban con un precio más bajo. Aunque hay que tener varias variables en cuenta podría inferirse por el gráfico que esto se cumple, y que debería estudiarse que tal vez las personas suelen hacer más reseñas en malas experiencias, lo que baja el precio del Airbnb.

****

**Análisis de Componentes principales**

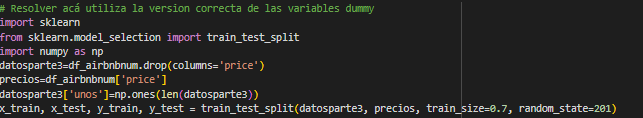
El análisis de componentes principales (PCA) muestra que los dos primeros componentes explican el 48% de la varianza total. Esto implica que una parte importante de la variabilidad en las variables originales se puede representar en solo dos dimensiones. No se observan patrones claros en el gráfico de dispersión de los componentes, lo que indica que las variables no tienen una fuerte diferenciación lineal. Los loadings revelan que el "precio" y la "disponibilidad" son las variables que más contribuyen a los componentes principales.

****

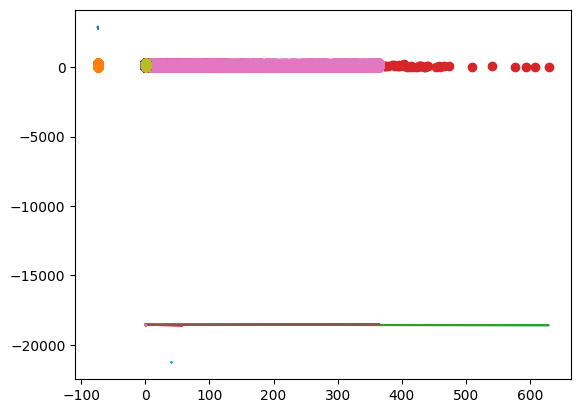
**Predicción**

Por último, se realizó una regresión lineal utilizando como variable dependiente a ‘price’, con los demás variables funcionando como variables independientes.

Se separó la muestra en un grupo de entrenamiento con una proporción del 70% de la muestra siendo utilizada para entrenar al modelo, para ello se utilizó el comando train\_test\_split y se aleatorizó con la semilla 201



Cuando se analizó utilizando los datos de entrenamiento se encontró un r2 de 0.5005057415402637, con un intercept de -18551.04 y un gráfico de comparación (entre la variable y y las variables x)



Por último, se realizó una predicción utilizando los valores que no fueron utilizados para realizar el modelo como prueba de predicción. Se realizó el r2 con el y estimado y los y reales resultando en un r2 de 0.4992684763881372 menor al original y un mean squared error MSE de 2134.273876944542. Lo que nos indica que el modelo no resulta adecuado para predecir las variables.