Data Storytelling & Methodology Document

– AirBnb, New York Analysis using Python

By – Shivansh singh, Aparna Sahu, Manish Matsaniya

A. Methodology Approach

1. Research Problem

    - For the past few months, Airbnb has seen a major decline in revenue due to the lockdown imposed during the pandemic.
    - Now after 2 years of devastating Covid pandemic, the restrictions knot have started losing, and people have started to travel more. Hence, Airbnb wants to make sure to be fully prepared for this change.

2. Business Understanding

    - Airbnb id an American company based in San Francisco, California. It operates an online market places for lodging, primarily home stays for vacation rental, and tourism activities. The platform can be reachable via its web sites and mobile application.
    - Being as online market place for hosting personal home stays and private apartments in the majority, and sue to this, company have basically two types of customers. One who host their place and the another who book the place for a particular time is the end consumer utilizing the hosted place.
    - Airbnb earn commission from both ends and hence Airbnb wants to sure about both the customers that are able to generate values from their business.
    - It also offered hosted place on the platform to provide the best services at reasonable prices and lookout for the best technology to ease out the booking process for the end consumer without hassle.

3. Type of Data required to Analyze

    - The decline in the revenue of the company could have many reasons, either platform service is not so good or home /apartment service and reviews are not so impressive. It could also happen that, Airbnb home locations are not on the right place, might be too far away from the Airports or any private travel agency. It could also happens that prices are too high to afford and too low to attract or impress customers due to the negative mindset of having bad services with cheap homes/apartments.

4. How Data has acquired ? (Assumption)

    - The Provided data is captured from the CRM tool used by Airbnb to manage their customers that are hosting sites on their platform.
    - The reviews provided in the data frame are assumed to be positive as it is not mentioned whether they are in negative or positive reviews.

5. Whom we are presenting ?

    - **Data Analysis Managers :** These people manages the data directly for the processes and their technical expertise.
    - **Lead Data Analyst :** It looks after the entire team of the data and business analyst and it should have some technical taste.
    - **Head of Acquisitions and Operations, NYC:** It looks after the property and hosts acquisition and operations. Acquisition of the best properties, price negotiating and service negotiating for properties offers, all these work fall under this head role.
    - **Head of User Experience, NYC :** It looks after the customer preference and handles the properties listed on the website and the Airbnb app. Basically, it tries to optimize the order of property listing in certain neighborhoods and cities in order to get property the optimal amount of traction.

6. Recommendations –

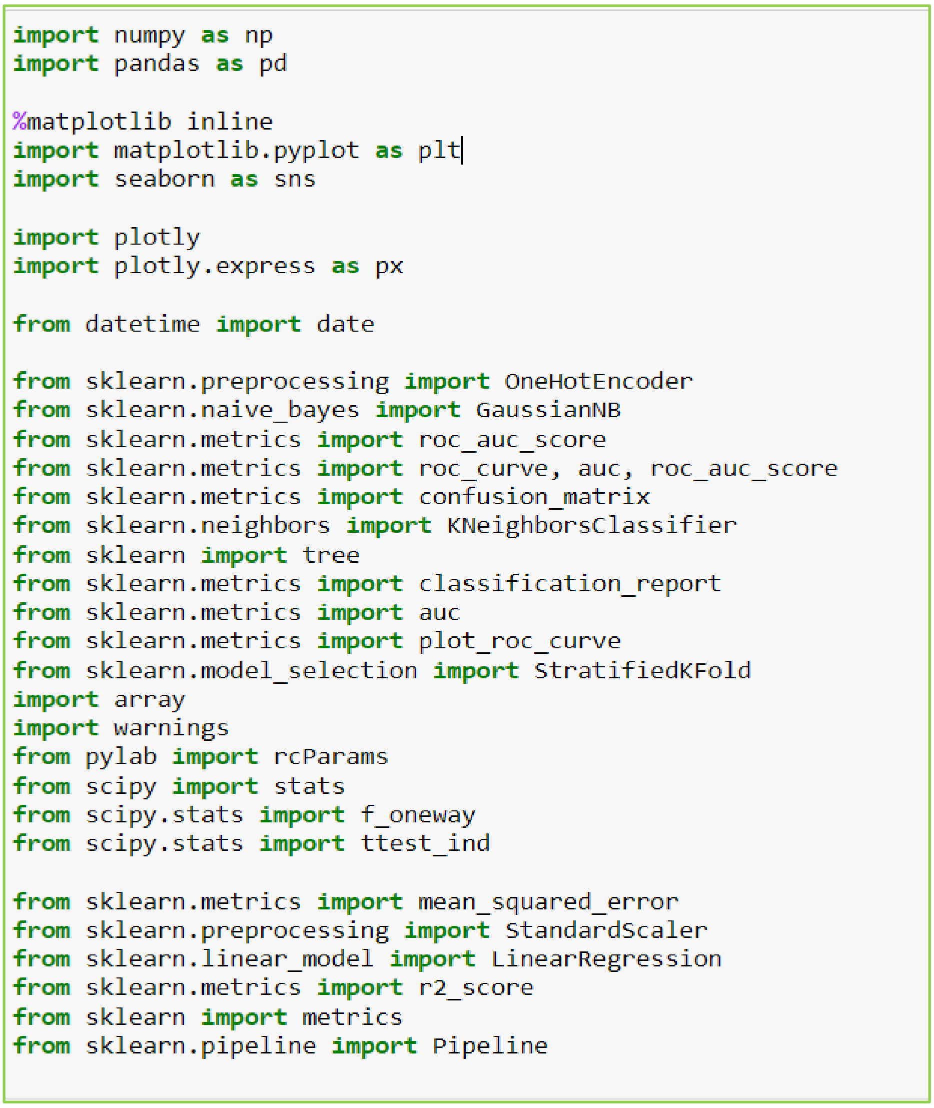
- One to one interaction with some property owners in the Bronx, the Queens and the Staten Island to identify their challenges for being fully functional for a maximum number of days in a year and allow a booking of more than 10 days of minimum nights stays.
- Create some sort of interactions between the Top 5 Hosts to share their experiences with the rest of the community. It might be on the YouTube Channel, or their Online websites, and this could be for better improvement and value-generating ideas.
- Provides some discount commission rates to property owners on keeping the minimum night stay booking window for more than 10 days and property functional for a maximum number of days in a year.

B. Method of Analysis along with code :

1. Data Understanding and Preparation –

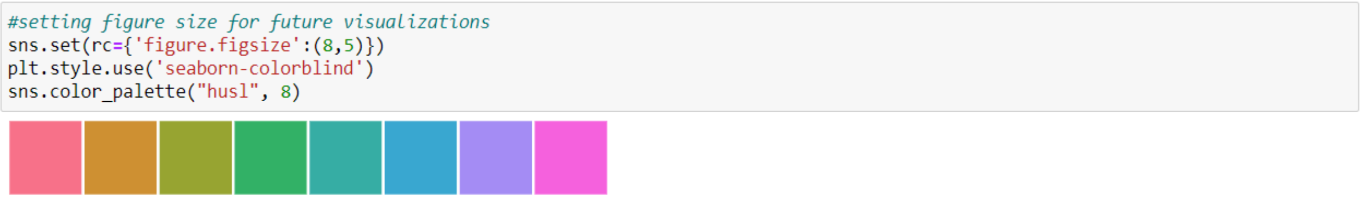
- Before we start the basic understanding of the data, lets import relevant libraries available in the Python.

# Importing Libraries –



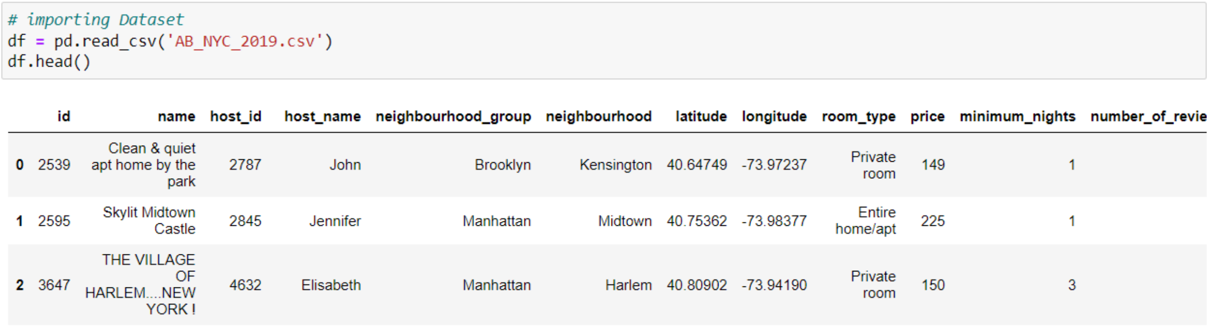
```
import numpy as np
import pandas as pd

%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns

import plotly
import plotly.express as px

from datetime import date

from sklearn.preprocessing import OneHotEncoder
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import roc_auc_score
from sklearn.metrics import roc_curve, auc, roc_auc_score
from sklearn.metrics import confusion_matrix
from sklearn.neighbors import KNeighborsClassifier
from sklearn import tree
from sklearn.metrics import classification_report
from sklearn.metrics import auc
from sklearn.metrics import plot_roc_curve
from sklearn.model_selection import StratifiedKFold
import array
import warnings
from pylab import rcParams
from scipy import stats
from scipy.stats import f_oneway
from scipy.stats import ttest_ind

from sklearn.metrics import mean_squared_error
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
from sklearn import metrics
from sklearn.pipeline import Pipeline
```

We also set the graph size default and color style use in future visualizations :



```
#setting figure size for future visualizations
sns.set(rc={'figure.figsize':(8,5)})
plt.style.use('seaborn-colorblind')
sns.color_palette("husl", 8)
```

We started with Understanding the Data in hand provided by running basic functions to load and interpret the Variables, data types of the variables, dimensions, and size of the data frame.



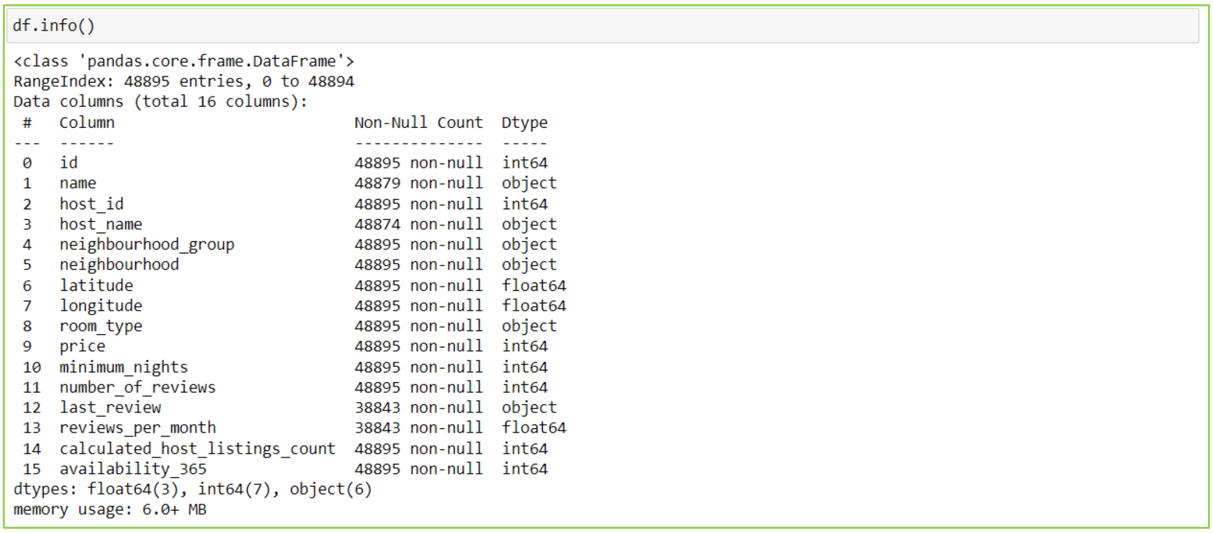
```
# importing Dataset
df = pd.read_csv('AB_NYC_2019.csv')
df.head()
```

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_revie
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kensington	40.64749	-73.97237	Private room	149	1	
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown	40.75362	-73.98377	Entire home/apt	225	1	
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harlem	40.80902	-73.94190	Private room	150	3	

# Dimensions

df.shape

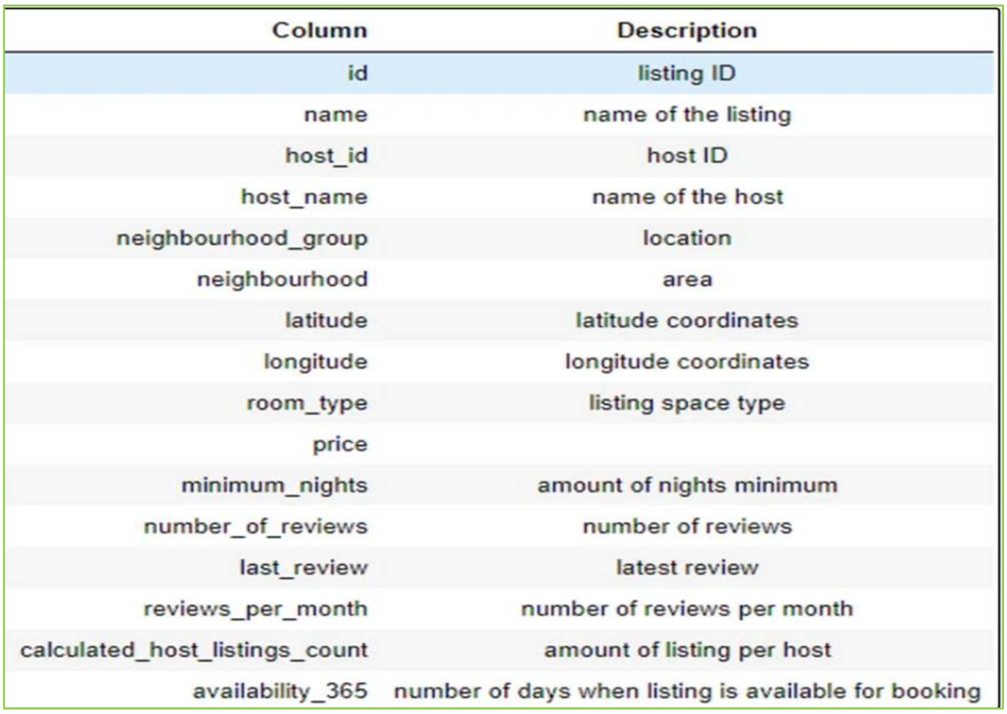
# Data – Types



```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48895 entries, 0 to 48894
Data columns (total 16 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   id                              48895 non-null  int64  
 1   name                            48879 non-null  object 
 2   host_id                         48895 non-null  int64  
 3   host_name                       48874 non-null  object 
 4   neighbourhood_group             48895 non-null  object 
 5   neighbourhood                   48895 non-null  object 
 6   latitude                        48895 non-null  float64
 7   longitude                       48895 non-null  float64
 8   room_type                       48895 non-null  object 
 9   price                           48895 non-null  int64  
 10  minimum_nights                  48895 non-null  int64  
 11  number_of_reviews               48895 non-null  int64  
 12  last_review                     38843 non-null  object 
 13  reviews_per_month               38843 non-null  float64
 14  calculated_host_listings_count  48895 non-null  int64  
 15  availability_365                48895 non-null  int64  
dtypes: float64(3), int64(7), object(6)
memory usage: 6.0+ MB
```

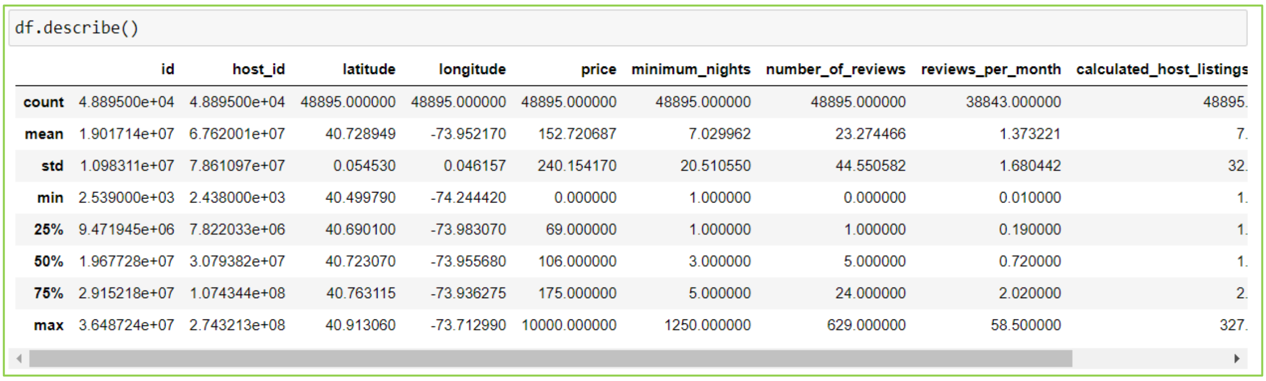
2. Variables in the data frame:



Column	Description
id	listing ID
name	name of the listing
host_id	host ID
host_name	name of the host
neighbourhood_group	location
neighbourhood	area
latitude	latitude coordinates
longitude	longitude coordinates
room_type	listing space type
price	
minimum_nights	amount of nights minimum
number_of_reviews	number of reviews
last_review	latest review
reviews_per_month	number of reviews per month
calculated_host_listings_count	amount of listing per host
availability_365	number of days when listing is available for booking

The above understandings lead us to perform basic **Numeric and Categorical analysis** in depth by using the following function along with some basic wear and tear.

# Numeric Analysis

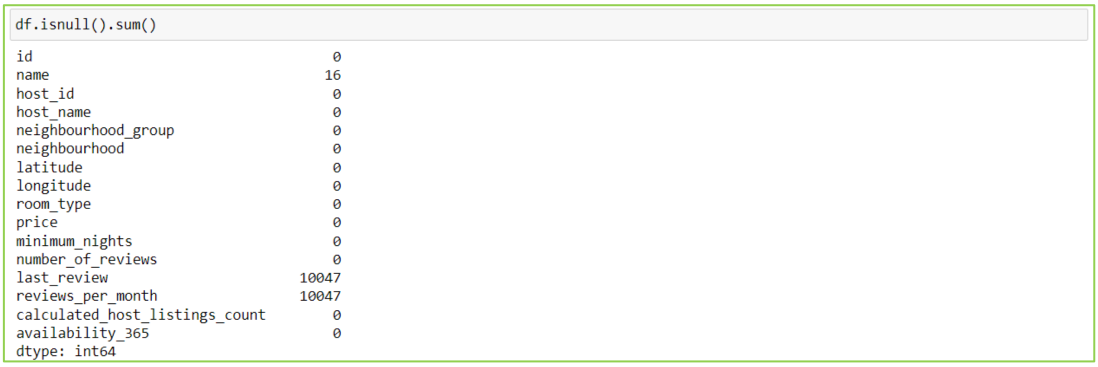


```
df.describe()
```

	id	host_id	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_listings
count	4.889500e+04	4.889500e+04	48895.000000	48895.000000	48895.000000	48895.000000	48895.000000	38843.000000	48895.
mean	1.901714e+07	6.762001e+07	40.728949	-73.952170	152.720687	7.029962	23.274466	1.373221	7.
std	1.098311e+07	7.861097e+07	0.054530	0.046157	240.154170	20.510550	44.550582	1.680442	32.
min	2.539000e+03	2.438000e+03	40.499790	-74.244420	0.000000	1.000000	0.000000	0.010000	1.
25%	9.471945e+06	7.822033e+06	40.690100	-73.983070	69.000000	1.000000	1.000000	0.190000	1.
50%	1.967728e+07	3.079382e+07	40.723070	-73.955680	106.000000	3.000000	5.000000	0.720000	1.
75%	2.915218e+07	1.074344e+08	40.763115	-73.936275	175.000000	5.000000	24.000000	2.020000	2.
max	3.648724e+07	2.743213e+08	40.913060	-73.712990	10000.000000	1250.000000	629.000000	58.500000	327.

3. Handling Missing Values and Outliers :

- ❖ Then we moved to handle missing values and outliers in the data frame.



```
df.isnull().sum()
```

```
id                                    0
name                                 16
host_id                               0
host_name                             0
neighbourhood_group                   0
neighbourhood                         0
latitude                              0
longitude                             0
room_type                             0
price                                 0
minimum_nights                        0
number_of_reviews                     0
last_review                       10047
reviews_per_month                 10047
calculated_host_listings_count        0
availability_365                      0
dtype: int64
```

- ❖ Starting with the missing values, we identified two columns having an equal percentage of missing values which were last_review and reviews_per_month of around 20.56%. And also, the other two columns had quite minimal missing values which were host_name of 0.4% and name of the place of 0.3%.

- Then we analyzed the values missing in last_review and reviews_per_month carrying NaN values on purpose, meaning they are not missing at random as these hosted sites/places have not received any reviews from the customers. Hence, these places would be least preferred by the future customers and would also be facing bad business from our side.

- Then we identified that we have 16 places and 21 hostnames that are missing and then we cross-check them with the help of their IDs to verify whether they are missing at random or by chance.

- After analyzing, it seems like these values (hostname and their place name) are missing by chance hence we need to collect this information from the Host acquisition and operations team. But for now, we left these rows blank.

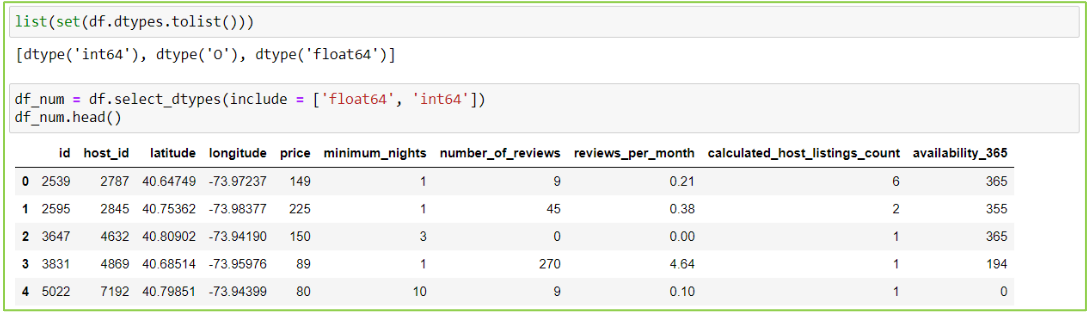
- Finally, we just imputed the missing values of reviews_per_month with a 0.

# Checking missing values percentages

```
def null_values(df):
return round((df.isnull().sum()/len(df)*100).sort_values(ascending = False),2)
null_values(df)
```

Post analyzing and treating the missing values accordingly we treated the spread in the data frame i.e. outliers. Below is the code we used to identify the spread of the outliers.

# Extracting Numeric columns:



```
list(set(df.dtypes.tolist()))
```

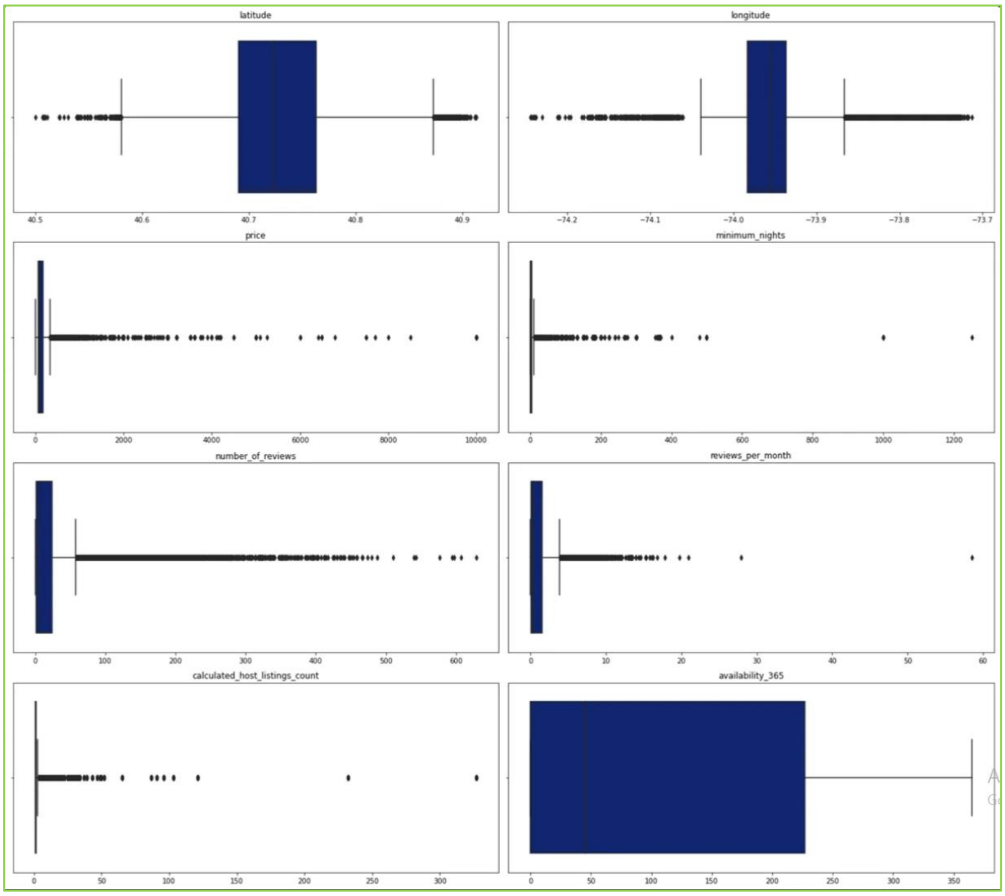
```
[dtype('int64'), dtype('O'), dtype('float64')]
```

```
df_num = df.select_dtypes(include = ['float64', 'int64'])
df_num.head()
```

	id	host_id	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_listings_count	availability_365
0	2539	2787	40.64749	-73.97237	149	1	9	0.21	6	365
1	2595	2845	40.75362	-73.98377	225	1	45	0.38	2	355
2	3647	4632	40.80902	-73.94190	150	3	0	0.00	1	365
3	3831	4869	40.68514	-73.95976	89	1	270	4.64	1	194
4	5022	7192	40.79851	-73.94399	80	10	9	0.10	1	0

# Plotting the spread of outliers:

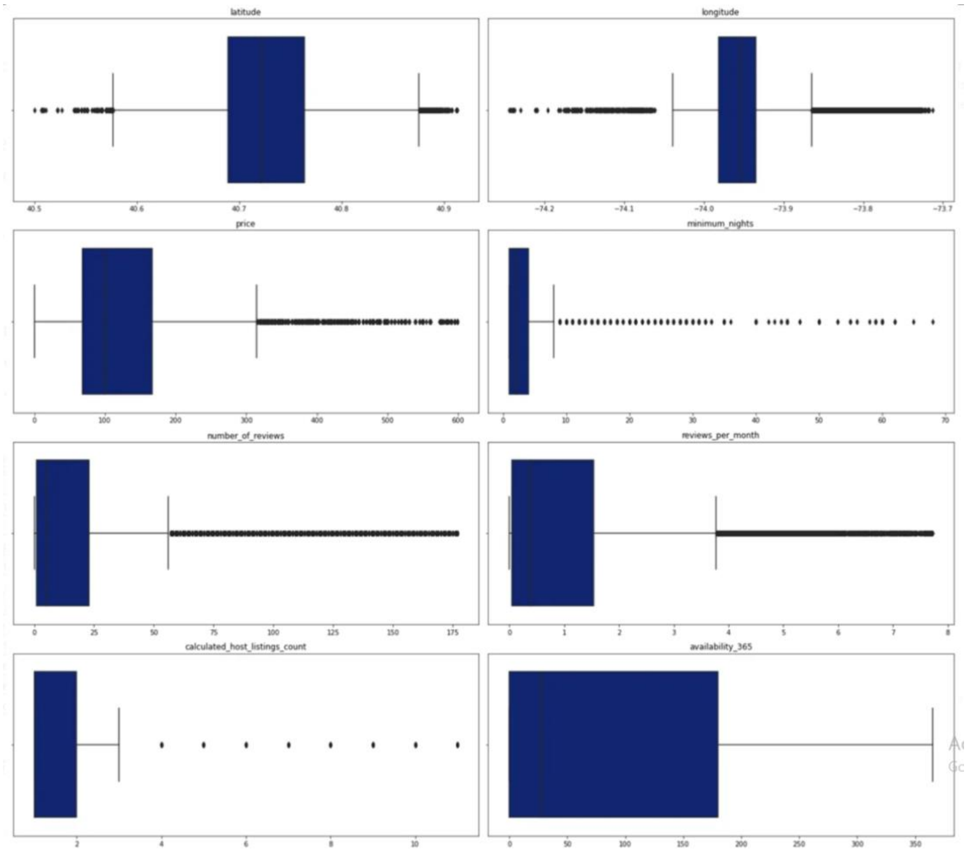
```
plt.figure(figsize=([20,22]))
for n,col in enumerate(int_cols):
plt.subplot(5,2,n+1)
sns.boxplot(df[col], orient = “h”)
plt.xlabel(“”)
plt.ylabel(“”)
plt.title(col)
plt.tight_layout()
```



The method we used to treat them was by capping them by 10% Below is the code for the same.

```
# Capping (statistical) outliers
# outlier treatment for price:
Q1 = df.price.quantile(0.10)
Q3 = df.price.quantile(0.90)
IQR = Q3 — Q1
df = df[(df.price >= Q1–1.5*IQR) & (df.price <= Q3 + 1.5*IQR)]
# outlier treatment for minimum_nights:
Q1 = df.minimum_nights.quantile(0.10)
Q3 = df.minimum_nights.quantile(0.90)
IQR = Q3 — Q1
df = df[(df.minimum_nights >= Q1–1.5*IQR) & (df.minimum_nights <= Q3 + 1.5*IQR)]
# outlier treatment for minimum_nights:
Q1 = df.number_of_reviews.quantile(0.10)
Q3 = df.number_of_reviews.quantile(0.90)
IQR = Q3 — Q1
df = df[(df.number_of_reviews >= Q1–1.5*IQR) & (df.number_of_reviews <= Q3 + 1.5*IQR)]
# outlier treatment for reviews_per_month:
Q1 = df.reviews_per_month.quantile(0.10)
Q3 = df.reviews_per_month.quantile(0.90)
```

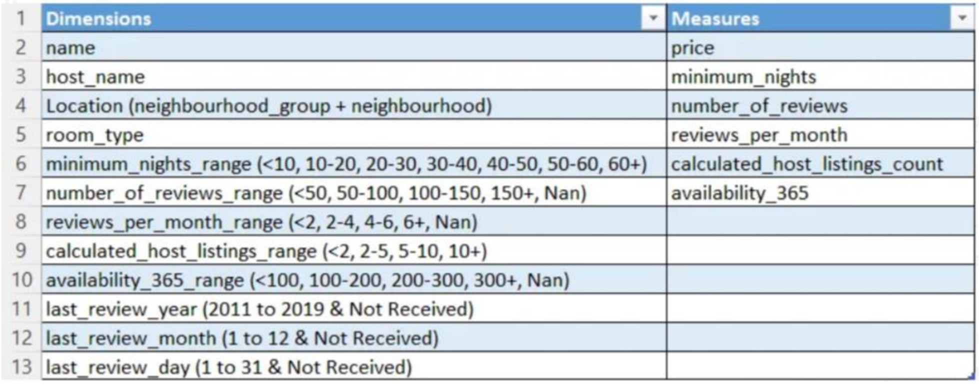
```
IQR = Q3 — Q1
df = df[(df.reviews_per_month >= Q1–1.5*IQR) & (df.reviews_per_month <= Q3 + 1.5*IQR)]
# outlier treatment for calculated_host_listings_count:
Q1 = df.calculated_host_listings_count.quantile(0.10)
Q3 = df.calculated_host_listings_count.quantile(0.90)
IQR = Q3 — Q1
df = df[(df.calculated_host_listings_count >= Q1–1.5*IQR) &
(df.calculated_host_listings_count <= Q3 + 1.5*IQR)]
```



Looks like we were able to manage the outliers, enough to analyze the information in EDA.

4. Feature Selection / Engineering

- The most important step of our Data Preprocessing was to convert some of the numeric features into categorical variables by creating bins of them. Yet post-conversion, we kept the numeric one’s handy tool for analyzing. Below is the table of all the variables that were engineered for our further analyses.



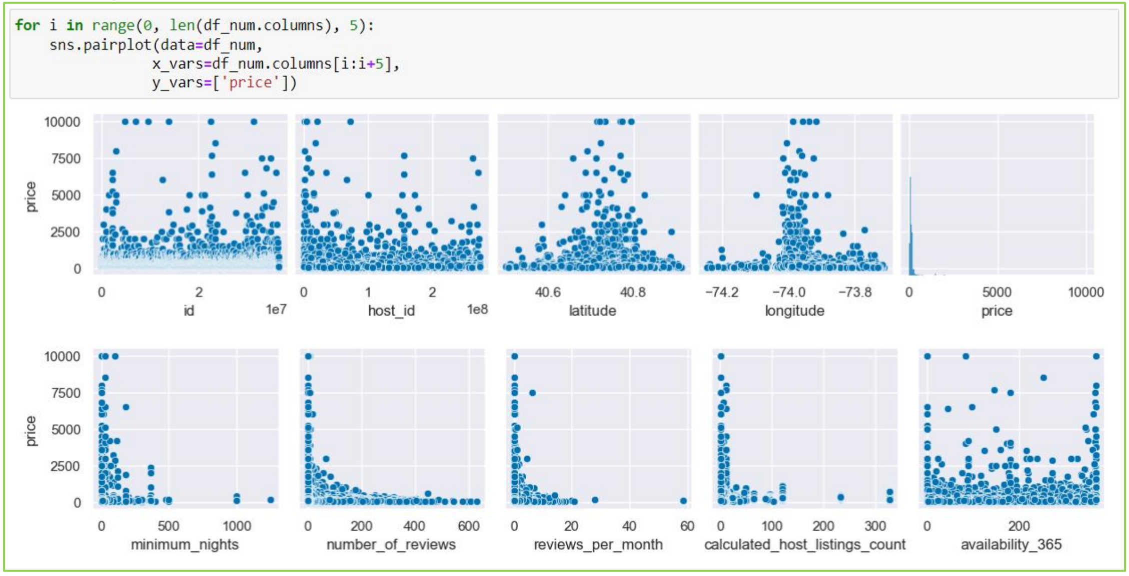
1	Dimensions	Measures
2	name	price
3	host_name	minimum_nights
4	Location (neighbourhood_group + neighbourhood)	number_of_reviews
5	room_type	reviews_per_month
6	minimum_nights_range (<10, 10-20, 20-30, 30-40, 40-50, 50-60, 60+)	calculated_host_listings_count
7	number_of_reviews_range (<50, 50-100, 100-150, 150+, Nan)	availability_365
8	reviews_per_month_range (<2, 2-4, 4-6, 6+, Nan)	
9	calculated_host_listings_range (<2, 2-5, 5-10, 10+)	
10	availability_365_range (<100, 100-200, 200-300, 300+, Nan)	
11	last_review_year (2011 to 2019 & Not Received)	
12	last_review_month (1 to 12 & Not Received)	
13	last_review_day (1 to 31 & Not Received)	

5. Analyzing Methods

a. Univariate Analysis:

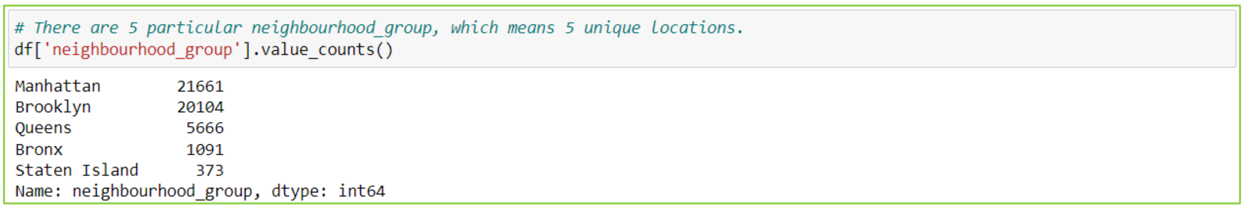
We started our general Univariate Analysis of Numeric and Categorical columns. For numeric columns, we used a Distribution plot from seaborn and for categorical columns, we used a Countplot from the same library seaborn. Below are the codes for the same.

# Plotting the Numeric Variables Distribution:



```
for i in range(0, len(df_num.columns), 5):
    sns.pairplot(data=df_num,
                 x_vars=df_num.columns[i:i+5],
                 y_vars=['price'])
```

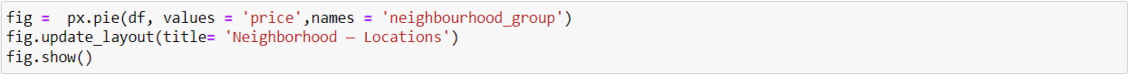
# Checking the count of Neighborhood Groups



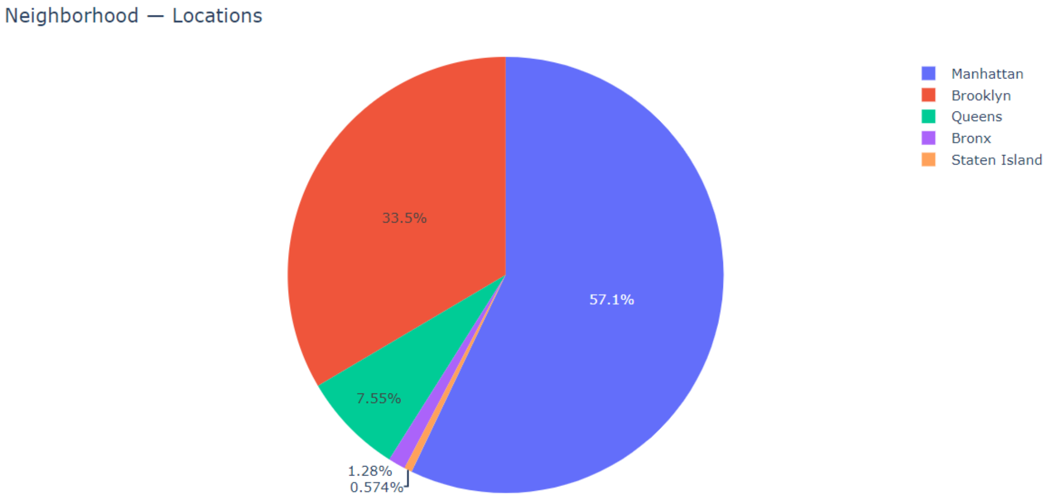
```
# There are 5 particular neighbourhood_group, which means 5 unique locations.
df['neighbourhood_group'].value_counts()
```

```
Manhattan        21661
Brooklyn         20104
Queens            5666
Bronx             1091
Staten Island      373
Name: neighbourhood_group, dtype: int64
```

# Plotting the count of Neighborhood Groups



```
fig =  px.pie(df, values = 'price',names = 'neighbourhood_group')
fig.update_layout(title= 'Neighborhood — Locations')
fig.show()
```

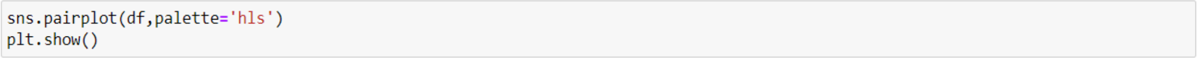


Similarly we used the above countplot code for the rest of the categorical plots created.

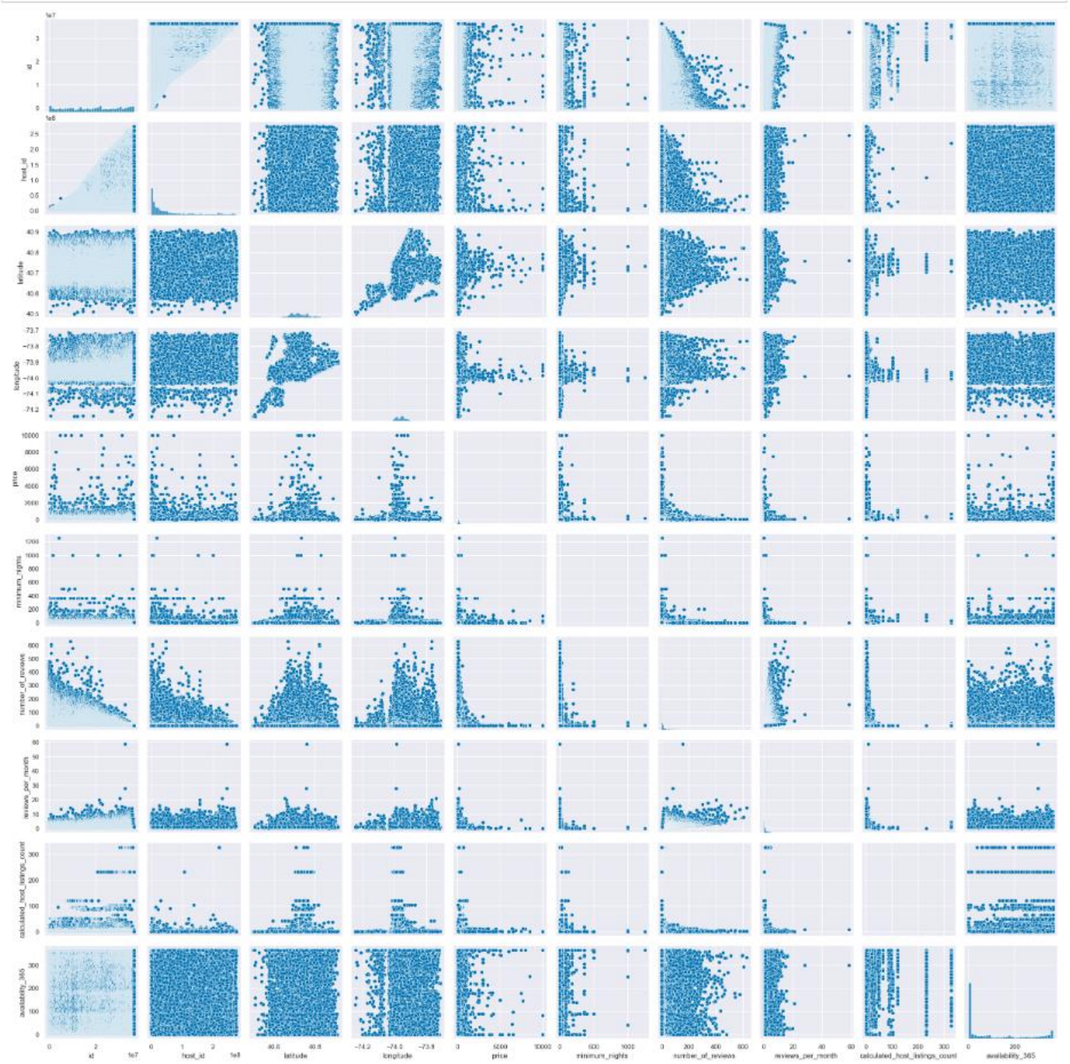
b. Bi-Multivariate Analysis:

Here we first plotted a pairplot of all the numeric columns using seaborn library in Python itself. Below is the code for the same.

# Plotting the pairplot



```
sns.pairplot(df,palette='hls')
plt.show()
```

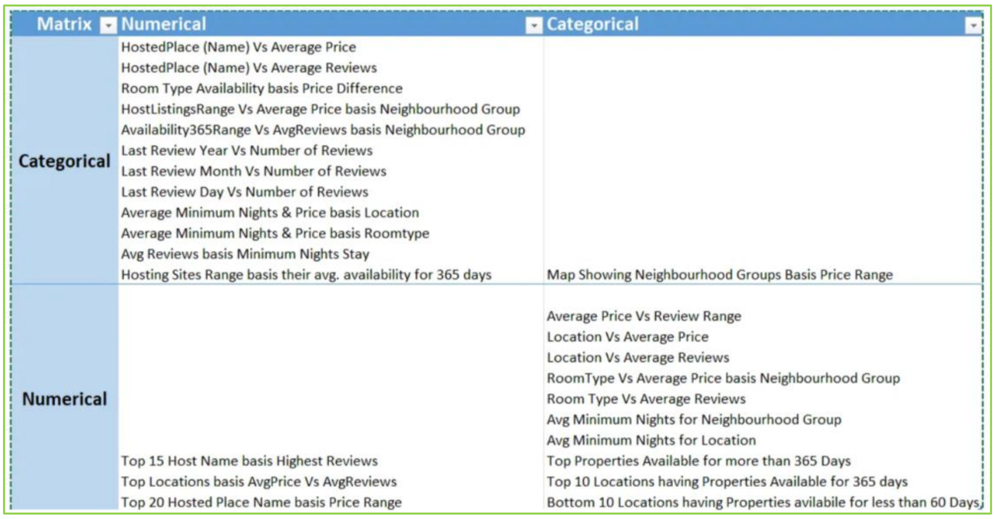


6. Matrix used for Analysis

In order to measure our analysis, we created a 2x2 Matrix to provide us a direction while creating graphs using different Dimensions and Measures. This matrix involved the values needed to create the graphs with the combinations of,

- Categorical & Numerical
- Categorical & Categorical
- Numerical & Numerical
- Numerical & Categorical

This turns out to be a road map for us, which helps in identifying which all dimensions and measures have been consolidated to get the insights from the data. Below is the Matrix.



Matrix	Numerical	Categorical
Categorical	HostedPlace (Name) Vs Average Price HostedPlace (Name) Vs Average Reviews Room Type Availability basis Price Difference HostListingsRange Vs Average Price basis Neighbourhood Group Availability365Range Vs AvgReviews basis Neighbourhood Group Last Review Year Vs Number of Reviews Last Review Month Vs Number of Reviews Last Review Day Vs Number of Reviews Average Minimum Nights & Price basis Location Average Minimum Nights & Price basis Roomtype Avg Reviews basis Minimum Nights Stay Hosting Sites Range basis their avg. availability for 365 days	Map Showing Neighbourhood Groups Basis Price Range
Numerical	Top 15 Host Name basis Highest Reviews Top Locations basis AvgPrice Vs AvgReviews Top 20 Hosted Place Name basis Price Range	Average Price Vs Review Range Location Vs Average Price Location Vs Average Reviews RoomType Vs Average Price basis Neighbourhood Group Room Type Vs Average Reviews Avg Minimum Nights for Neighbourhood Group Avg Minimum Nights for Location Top Properties Available for more than 365 Days Top 10 Locations having Properties Available for 365 days Bottom 10 Locations having Properties avilabile for less than 60 Days

7. Evaluation of Methods

The above matrix was evaluated at every step by creating relevant questions to see what we are trying to extract from the raw data. More importantly, to extract the relevant information that we want to recommend to our target audience. Below is the list of some questions that we crated to drive the above matrix for creating graphs.

Evaluating Questions:

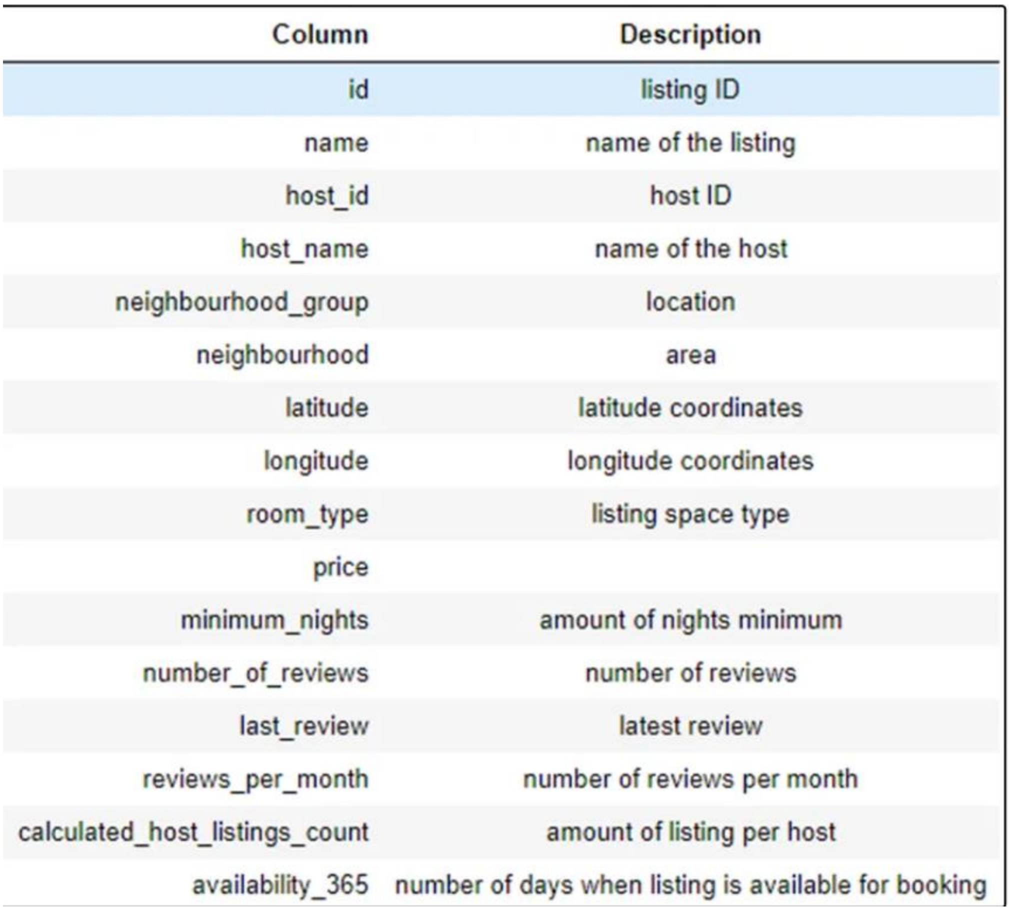
- ❖ Which locations are getting more traction?
- ❖ Which locations are price and review-sensitive?
- ❖ What are the pricing ranges preferred by end customers?
- ❖ What type of properties are preferred by the customers?
- ❖ Which properties are available for more days in a year and in which location?
- ❖ In what time period the properties have received more or fewer numbers of reviews?
- ❖ What are the most popular localities and properties in New York currently?
- ❖ Which properties and room types have more or less minimum night stay?
- ❖ How many sites are hosted by a single host and what are its success metrics?
- ❖ Which hosts have received better reviews?
- ❖ Which are the locations that are not performing well based on reviews and other parameters?
- ❖ Which are the room types that are not performing well?
- ❖ Which parameter makes the customer prefer the property and provide a review?
- ❖ Is there any correlation between the prices and reviews or other parameters?
- ❖ Which location has properties functioning for more than 300 days in a year or less than 50 days?

C. **Findings & Insights**

a) Basic Data Interpretation

- There are 16 columns and 48895 rows in the data frame.
- There are 3 floats, 7 integers, and 6 objects data type values in the data frame.
- There seem to be many columns with missing values.
- Need to check the reason behind the missing values and some feature engineering needed too.

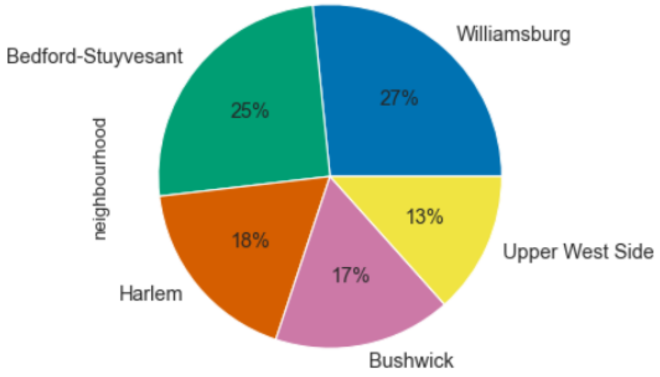
1. *Variables in the data frame:*



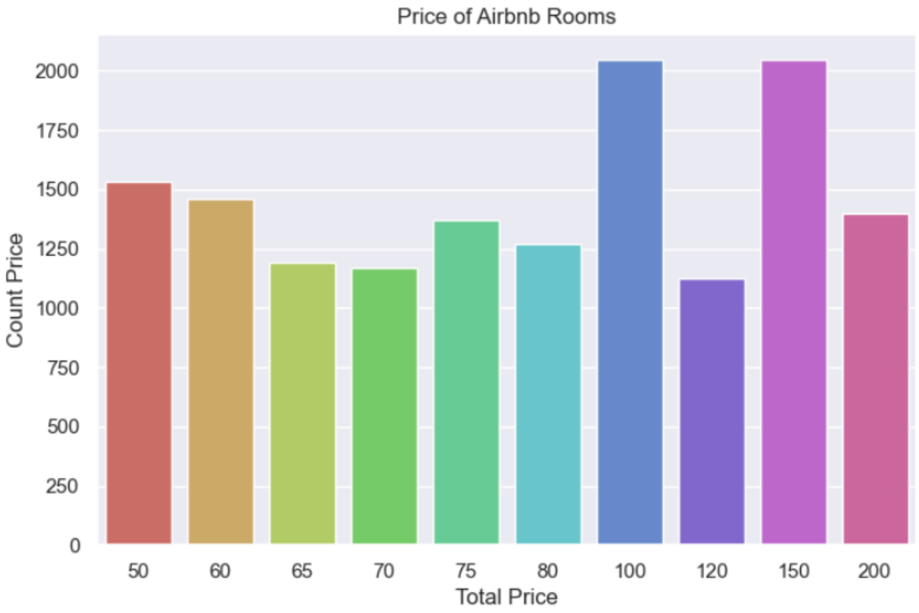
Column	Description
id	listing ID
name	name of the listing
host_id	host ID
host_name	name of the host
neighbourhood_group	location
neighbourhood	area
latitude	latitude coordinates
longitude	longitude coordinates
room_type	listing space type
price	
minimum_nights	amount of nights minimum
number_of_reviews	number of reviews
last_review	latest review
reviews_per_month	number of reviews per month
calculated_host_listings_count	amount of listing per host
availability_365	number of days when listing is available for booking

2. *Numerical Univariate Analysis:*
    - Latitude and Longitude obviously belong to New York City as we have the data from the same.
    - We can see prices starting from 0 dollars going up to 10 grand in dollars for hosting a place. Question: why is there a 0-dollar price for a hosted place on Airbnb and for which place?
    - There seems to be a huge variance in minimum_nights, number_of_reviews, reviews_per_month, and calculated_host_listings_count columns. Some things need to be checked in detail too under these columns.
    - Manhattan seems to be the friendliest neighborhood group and Williamsburg is the most known location underneath that.
    - Hillside Hotel is the top-hosted place which is the Entire home/apt room type.
    - This place is hosted by Michael.

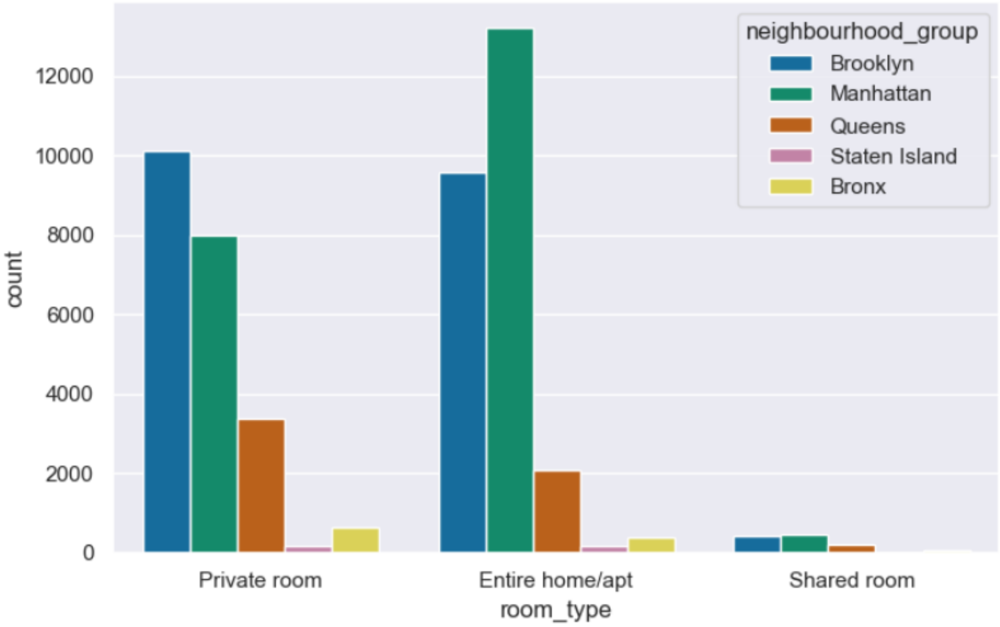
3. *Categorical Univariate Analysis:* **Findings:**

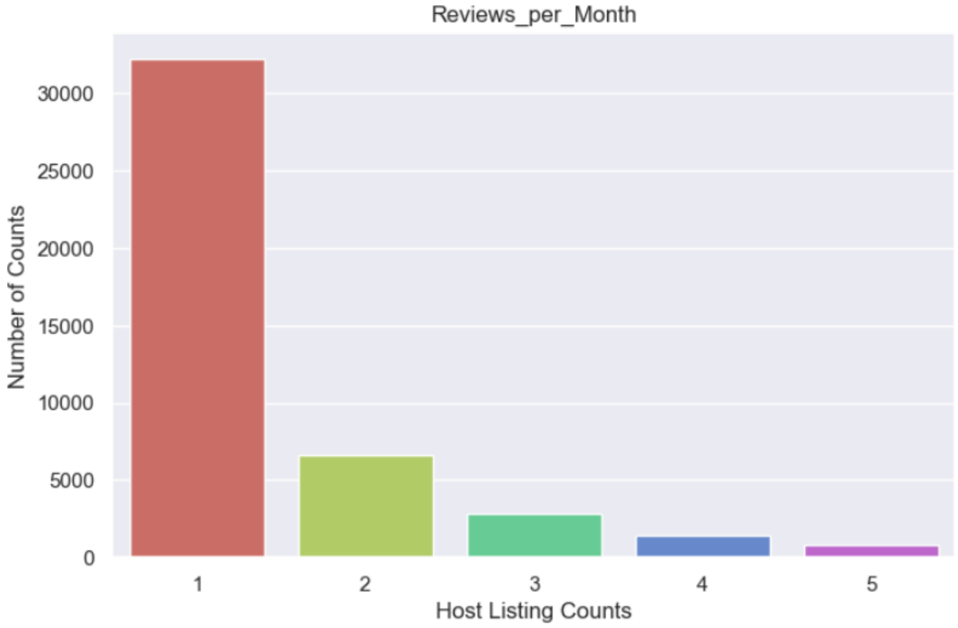


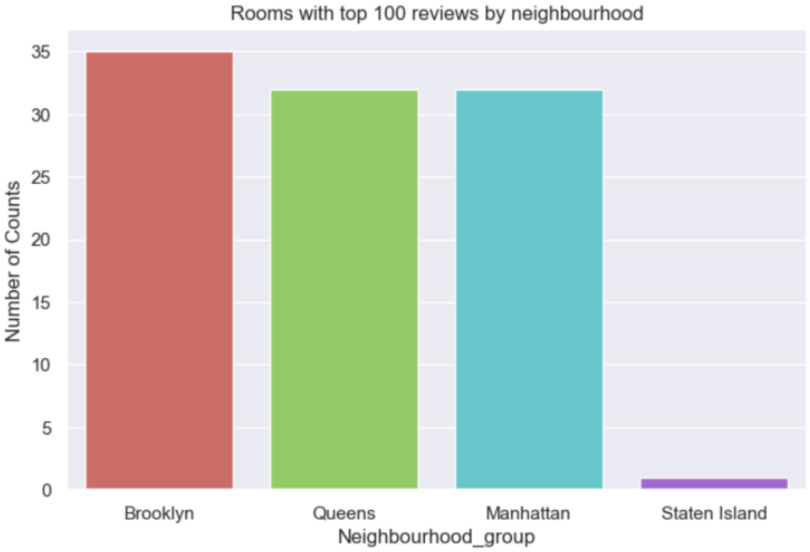
availability of neighbourhood types



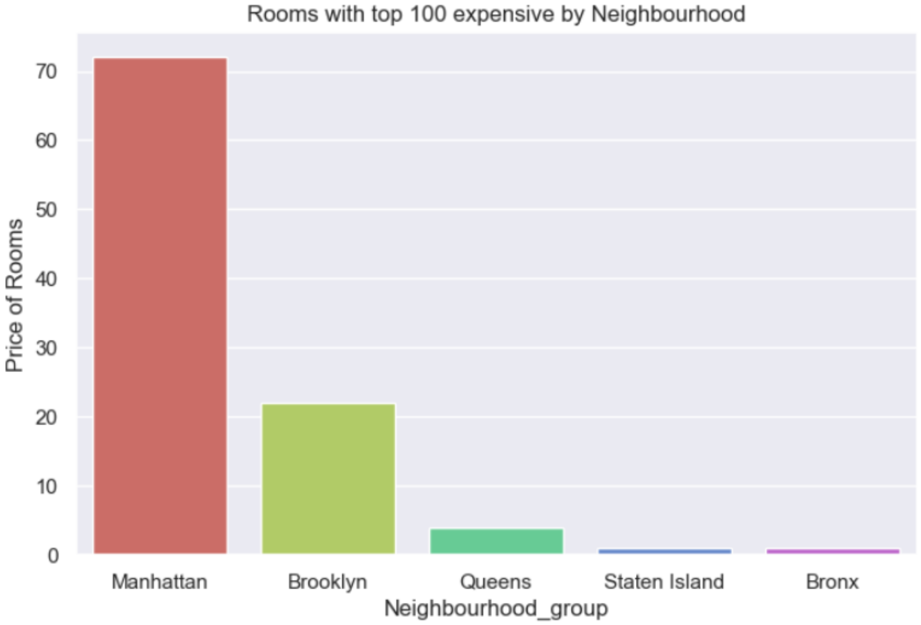
Almost 2k+ airbnb's has a price of 100 dollars and 150 dollars each respectively.

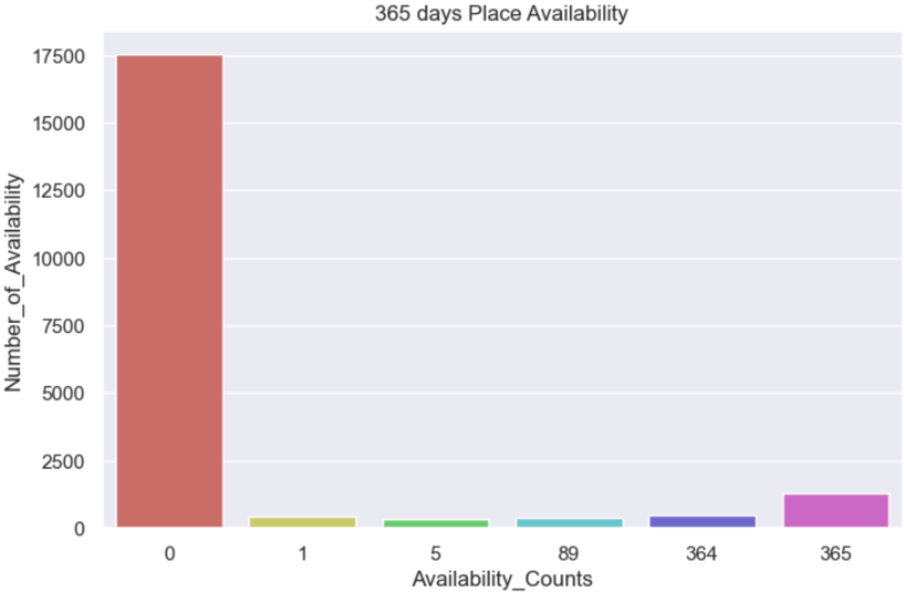


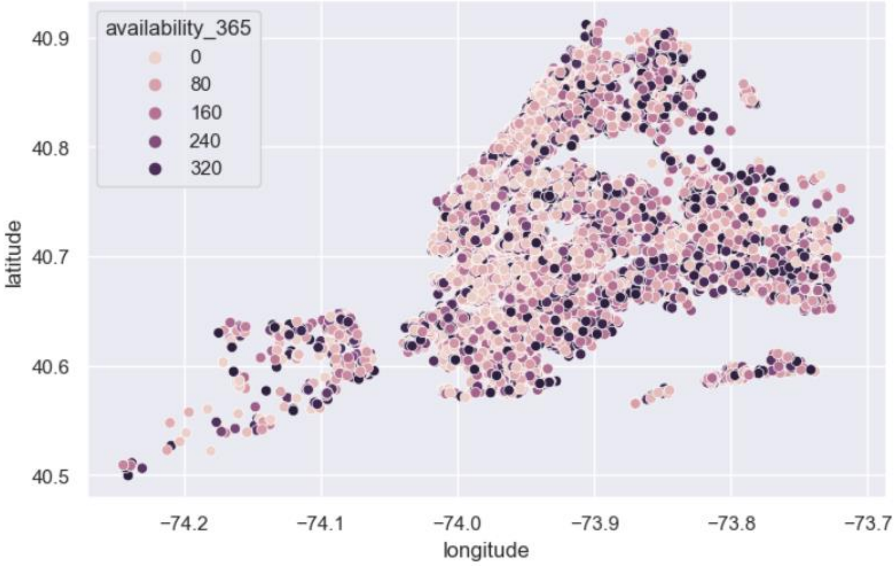


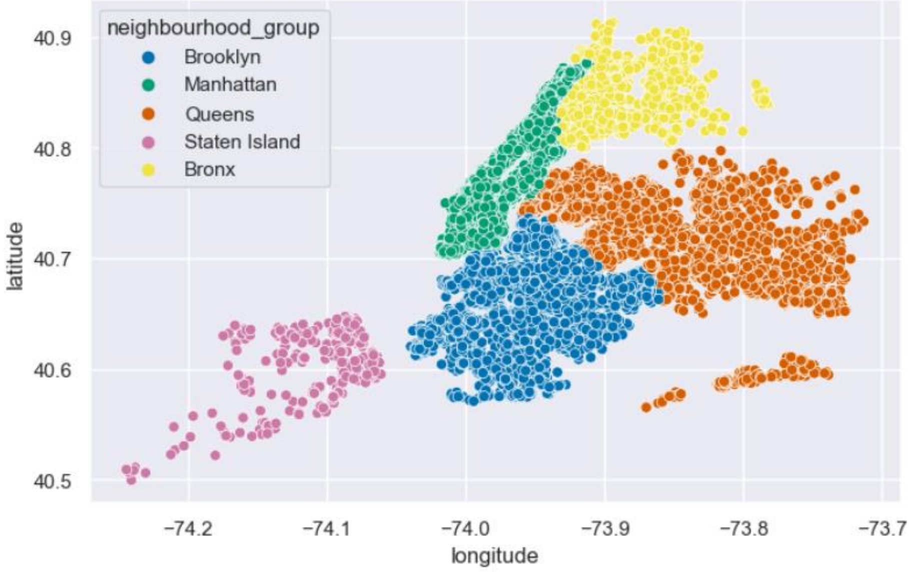


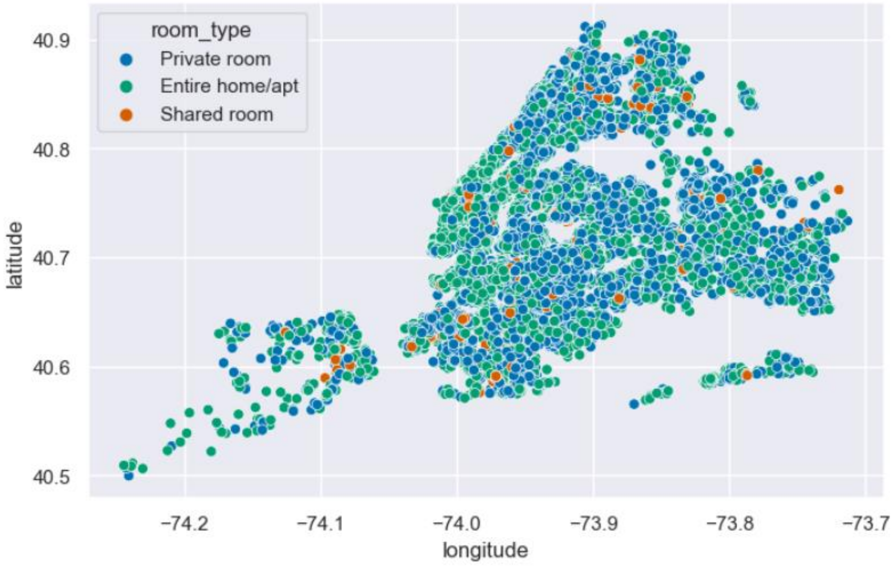
If we look at the top 100 airbnb's with number of reviews, Brooklyn has highest reviews followed by Queens and then Manhattan

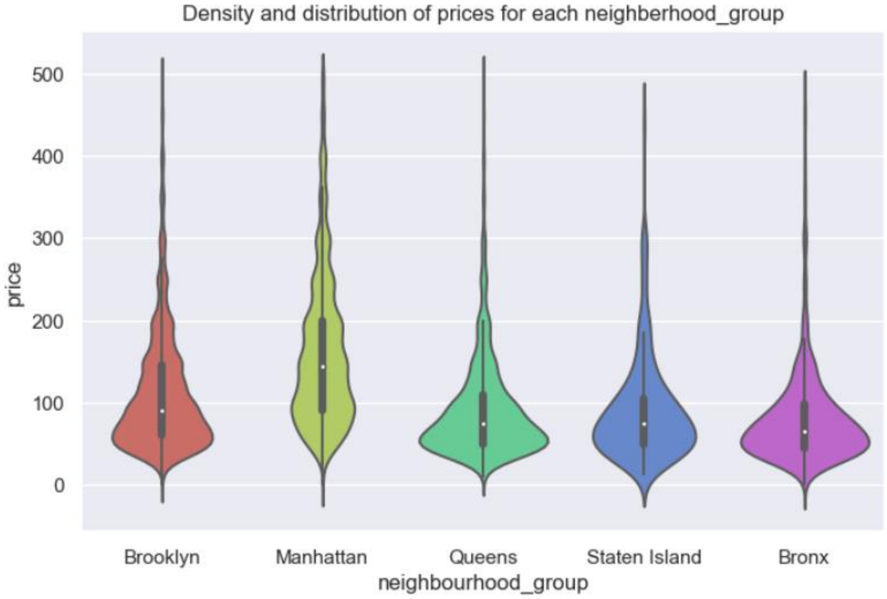



365 days Place Availability
Number_of_Availability
17500
15000
12500
10000
7500
5000
2500
0
0
1
5
89
364
365
Availability_Counts

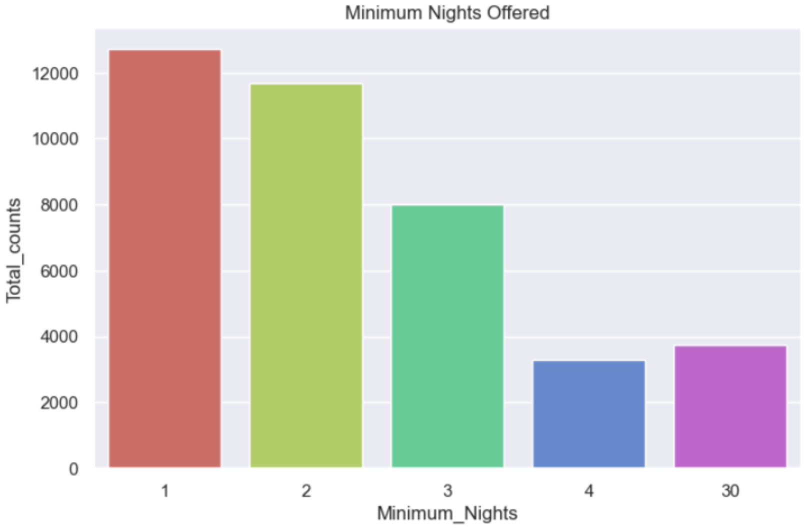

availability_365
0
80
160
240
320
latitude
40.9
40.8
40.7
40.6
40.5
−74.2
−74.1
−74.0
−73.9
−73.8
−73.7
longitude


neighbourhood_group
Brooklyn
Manhattan
Queens
Staten Island
Bronx
latitude
40.9
40.8
40.7
40.6
40.5
−74.2
−74.1
−74.0
−73.9
−73.8
−73.7
longitude





Manhattan airbnb's has the highest average price.



Important Findings:

- There seems to be no positive or any type of correlation between the numerical variables.

- Manhattan is the only Neighborhood in the Borough that lies in offering the Highest Price range properties on the platform followed by others with a Medium Price range on average. Prices offered above 120$ on average are a High Price, between 80$ to 120$, the Medium Price range, and less than 80$ to be considered Low Price range property.

- Having a high price range, Entire home/apt types of rooms are available for less than 100 days on average followed by Private rooms on an average of 105 days and Shared rooms around 155 days on average being the lowest in price.

- Manhattan has the highest number of places listed around more than 10 by a single host with an average price of 230 $ followed by Brooklyn with an average price of 108$. On the other hand, all the hosts have less than 2 properties listed in either of the Borough on an average price range between 80 $ to 170 $.

- Brooklyn has received the highest number of reviews based on the availability to stay open for more than 200 days in a year. This is followed by Staten Island and then the Bronx. On the other hand, there are some sites in Staten Island which are not open for a single day at all and hence could be the reason they have received very low reviews from the end consumer. **We need to check which are these places and what issues are they facing?**

- Majority of the **customers prefer a price range of 120$ to 130$ on average** for a stay. As most of them have provided a good number of reviews between this price range.

- **Michael, David, Alex, John, and Daniel are the Top 5 hosts** that seem to have received the highest number of reviews for their listed sites and have also sites listed with a High price range.

- Staten Island — Silver Lake, Staten Island — Richmondtown, Staten Island — Eltingville, Staten Island — Huguenot, and Brooklyn — Manhattan Beach are the Top 5 locations with Low Price ranges that have received the highest number of reviews on average being the lowest in the Price range. On the contrary, Queens — Neposit, Manhattan — NoHo, Manhattan — Tribeca, Staten Island — Willowbrook, and Manhattan — Flatiron District is the highest in the Price range and have received a low number of reviews.

- **“““WELCOME TO BROOKLYN””” PARK SIDE VIEW STUDIO APT””” , “““Oasis on the Park”””, “““HELLO BROOKLYN””” PARK SIDE VIEW NEWLY RENO APT”””, “““Comfy Home Away From Home/Multiple rooms”, “””LOVE BROOKLYN””” Newly Renovated Studio APT””” and “““Cozy Retreat””” in North Crown Heights”**”” are the Top 6 listed places that have received the highest number of reviews.

- On average Entire home/apt types **are preferred more by the customers** followed by Private rooms and then Shared Rooms. Mostly because they are also available for a higher number of minimum night’s stay window booking as compared to Private and Shared rooms.

- **“Modern Duplex — Central Chelsea!!!” in Manhattan-Chelsea, “Spacious & Bright 3BRs Near Subways, Parks, Shops” in Brooklyn-Cobble Hill, “NYC LUXURY3 BEDROOMS IN**

MIDTOWN EAST & GYM& BALCONY" in Manhattan-Murray Hill, "An Artist's Inspiration: Sun-Soaked Chelsea Loft" in Manhattan-Chelsea and "Upper West Side elegance. Riverside" in Manhattan-Upper West Side are the Top 5 hosted places with highest price offerings.

- **"Brooklyn-Williamsburg", "Brooklyn-Bedford-Stuyvesant", "Manhattan-Harlem", "Brooklyn-Bushwick" and "Manhattan-Upper West Side"** are some places providing the highest number of minimum nights window for bookmaking **Manhattan and Brooklyn** are the top neighborhoods in offering maximum-minimum nights stay.

- The average number of reviews started increasing exponentially after 2015–2016. And the majority of the customers provide a higher number of reviews either between the months of May till July or at the starting of the year which shows the higher booking window in a year.

- There are 5766 properties that are open for more than 300 days a year. Around 2286 of them are from Brooklyn followed by Manhattan of around 1947 properties. And on the other hand, the properties that stay open for less than 50 days a year belong to Queens or Staten Island.

- We can confirm that the greatest parameter for any customer to prefer a property and provide a review is having a maximum or minimum night stay window booking and their probability of being open for more days in a year to some extent.

Tools Used - Python is used for Data Understanding, Pre-processing, and general Univariate and Multivariate Analysis.

Thank You!