AirBNB Case Study

Methodology Document

By

Samanyu Ghose

Problem Statement

- Airbnb has seen a major decline in revenue due to pandemic.
- As the effect of pandemic has started to decline and the restrictions have started to lifting and people have started to travel more, Airbnb wants to make sure that it is fully prepared for the change
- The different leaders in Airbnb wants to understand some important insights based on various attributes in the dataset so as to increase the revenue

Data Analysis Steps

1. Reading and Understanding the data

   The dataset is in .csv format which has the following fields

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

a.Reading the data

To read the data,

```
# read the file using pandas
```

Out[2]:

```
In [2]: # read the file using pandas
airbnb=pd.read_csv(r'C:\Users\saman\Downloads/AB_NYC_2019.csv')

airbnb.head()
```

Out[2]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_revi
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	
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clinton Hill	40.68514	-73.95976	Entire home/apt	89	1	
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East Harlem	40.79851	-73.94399	Entire home/apt	80	10	

b.Inspecting the dataframe

i. Looking at shape of dataframe

```
In [3]: # Checking shape

        airbnb.shape

Out[3]: (48895, 16)

In [4]: # Checking datatypes

        airbnb.dtypes

Out[4]: id                                int64
        name                             object
        host_id                           int64
        host_name                        object
        neighbourhood_group              object
        neighbourhood                    object
        latitude                        float64
        longitude                       float64
        room_type                        object
        price                             int64
        minimum_nights                    int64
        number_of_reviews                 int64
        last_review                      object
        reviews_per_month               float64
        calculated_host_listings_count    int64
        availability_365                  int64
        dtype: object
```

It is showing 48895 rows and 16 columns. Also we see that there is no problem with the datatype. Text, categorical and Dates field have object type and numerical variables have int64 type

ii. Looking for more info into the dataset

```
In [5]: # checking null values

        airbnb.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
```

Here there are some missing values in name, host_name, last_review and reviews_per_month. Let's count the missing values

c. Data Cleaning

It is necessary to look into the data and clean for analysis. We will look for missing values and outliers

i. Looking for missing values :-

```
In [6]: # checking number of missing values

        airbnb.isnull().sum()

Out[6]: id                                    0
        name                                 16
        host_id                               0
        host_name                            21
        neighbourhood_group                   0
        neighbourhood                         0
        latitude                              0
        longitude                             0
        room_type                             0
        price                                 0
        minimum_nights                        0
        number_of_reviews                     0
        last_review                       10052
        reviews_per_month                 10052
        calculated_host_listings_count        0
        availability_365                      0
        dtype: int64

In [7]: # let's check percentage of missing values

        100*(round(airbnb.isnull().sum()/len(airbnb.index),2))

Out[7]: id                                   0.0
        name                                 0.0
        host_id                              0.0
        host_name                            0.0
        neighbourhood_group                  0.0
        neighbourhood                        0.0
        latitude                             0.0
        longitude                            0.0
        room_type                            0.0
        price                                0.0
        minimum_nights                       0.0
        number_of_reviews                    0.0
        last_review                         21.0
        reviews_per_month                   21.0
        calculated_host_listings_count       0.0
        availability_365                     0.0
        dtype: float64
```

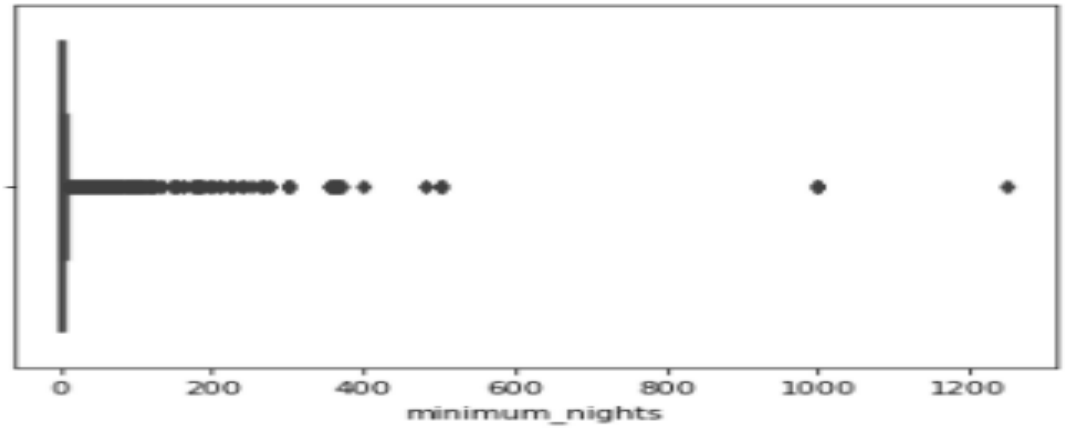
Since the percentage of missing values in reviews_per_month and last_review is not much, so we have not done any imputation.

Now we have to see the spread of data in categorical variables

```
In [8]: airbnb['room_type'].value_counts()

Out[8]: Entire home/apt    25409
        Private room       22326
        Shared room         1160
        Name: room_type, dtype: int64

In [9]: airbnb['neighbourhood_group'].value_counts()

Out[9]: Manhattan        21661
        Brooklyn         20104
        Queens            5666
        Bronx             1091
        Staten Island      373
        Name: neighbourhood_group, dtype: int64
```

ii. Looking for missing values

```
In [10]: # Checking for outliers in minimum_nights

         sns.boxplot(airbnb.minimum_nights)
         plt.show()
```



Since minimum_nights as outliers, so we have taken 0.99 values in and excluded the rest as outliers

Checked the data and limit it to .99 quantile

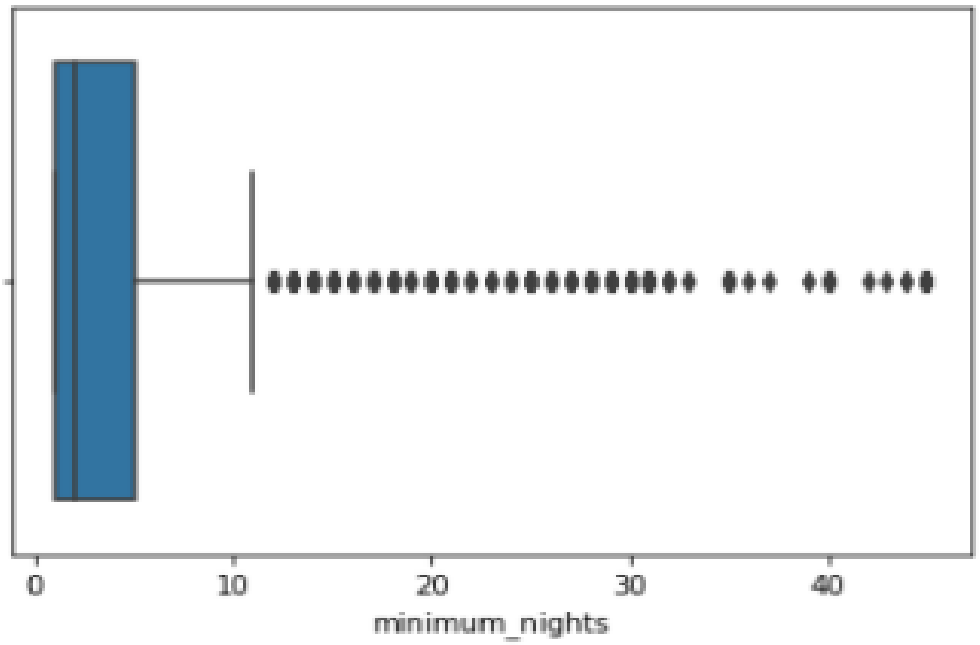
```
In [11]: airbnb.minimum_nights.quantile([0.90,0.91,0.92,0.93,0.94,0.95,0.96,0.97,0.98,0.99])

Out[11]: 0.90    28.0
         0.91    30.0
         0.92    30.0
         0.93    30.0
         0.94    30.0
         0.95    30.0
         0.96    30.0
         0.97    30.0
         0.98    30.0
         0.99    45.0
         Name: minimum_nights, dtype: float64

In [12]: airbnb=airbnb[airbnb['minimum_nights']<=45]
```

Now we have checked the data

```
In [13]: airbnb.shape
Out[13]: (48426, 16)
```

Checking again outlier to see if it is fixed now

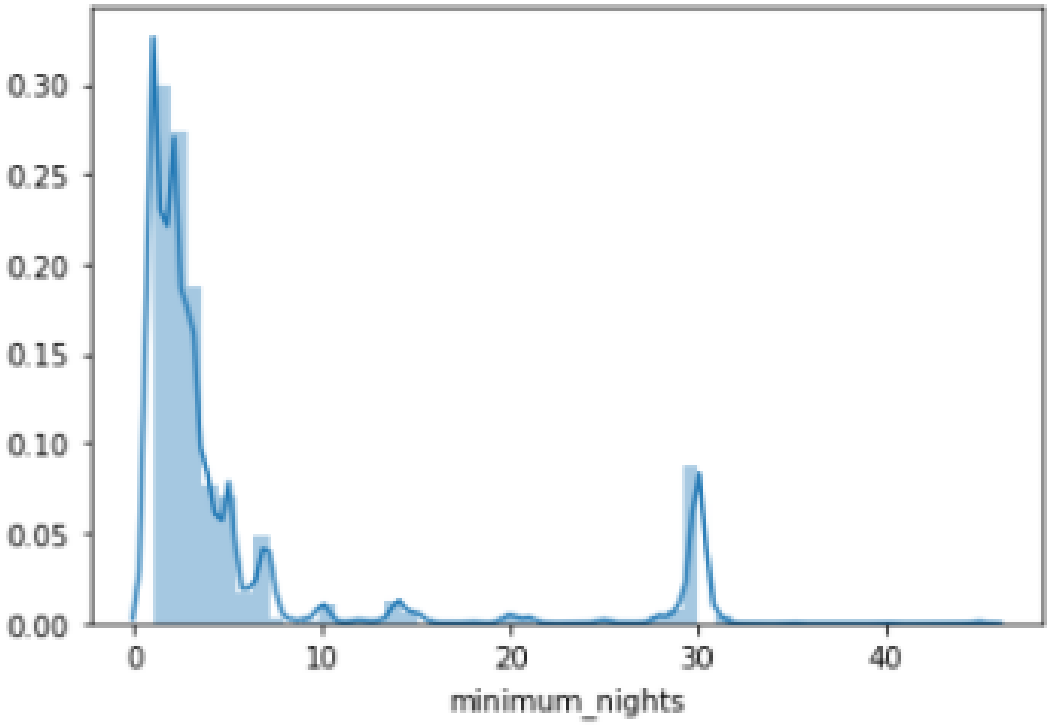
```
sns.boxplot(airbnb.minimum_nights)
plt.show()
```



Although it has outlier but they are continuous so not a big deal.

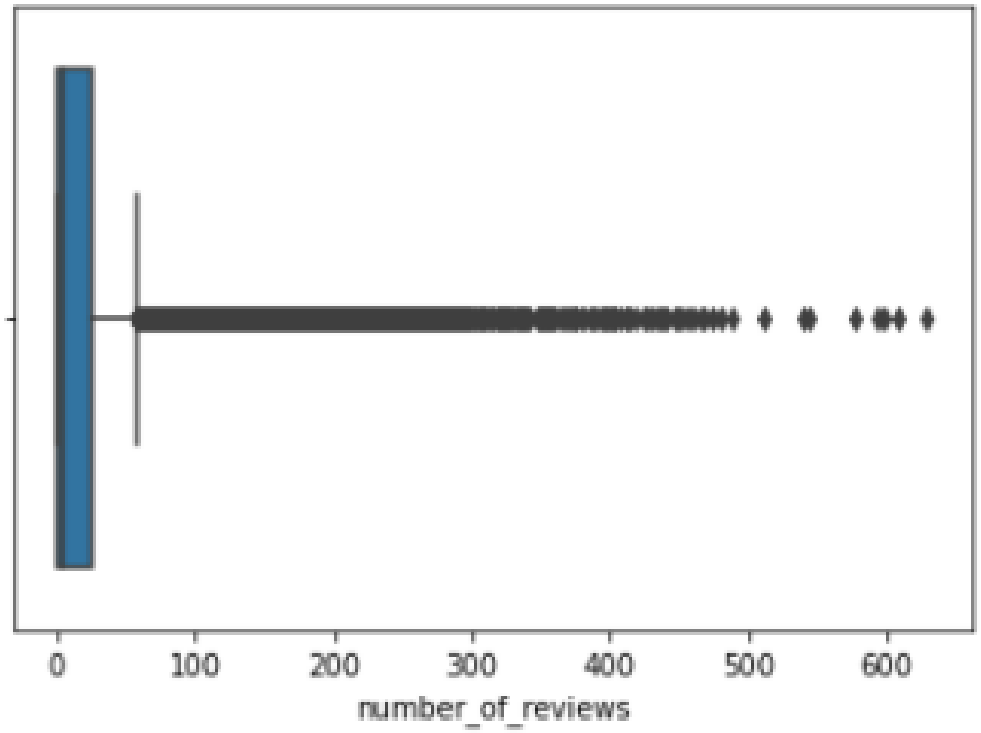
Looking at distribution of minimum_nights:

```
In [15]: sns.distplot(airbnb['minimum_nights'])
         plt.show()
```



Now checking number_of_reviews

```
sns.boxplot(airbnb['number_of_reviews'])
plt.show()
```



Since number_of_reviews has outliers, so we have taken 0.99 values in and excluded the rest as outliers

Checked the data and limit it to .99 quantile

```
airbnb.number_of_reviews.quantile([0.90,0.91,0.92,0.93,0.94,0.95,0.96,0.97,0.98,0.99])
```

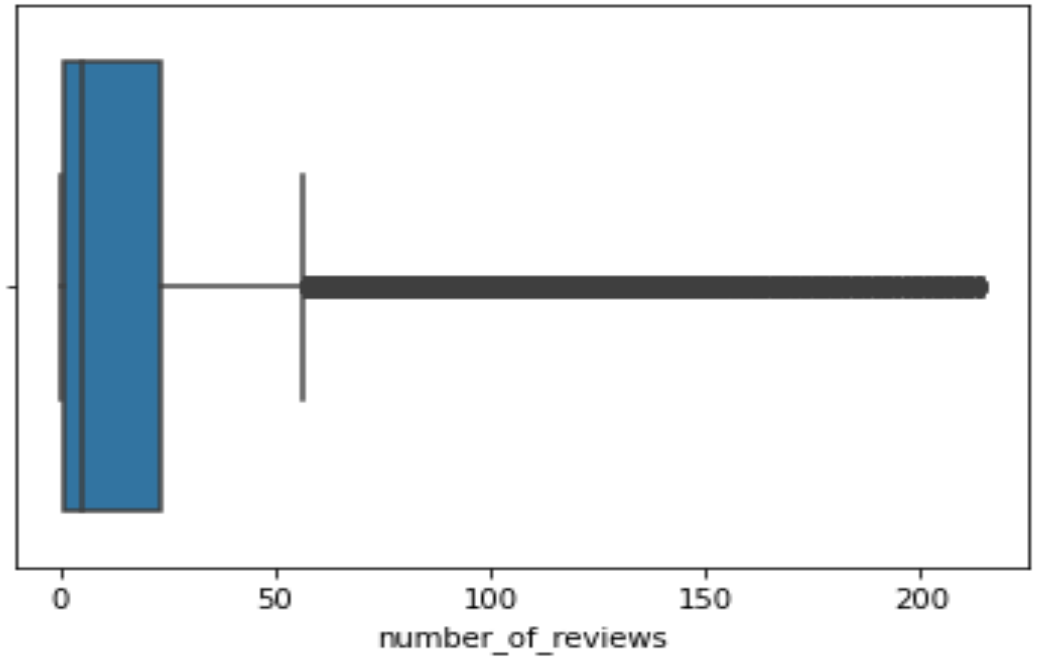
```
Out[17]: 0.90     71.0
         0.91     77.0
         0.92     84.0
         0.93     92.0
         0.94    102.0
         0.95    115.0
         0.96    129.0
         0.97    146.0
         0.98    172.0
         0.99    214.0
         Name: number_of_reviews, dtype: float64
```

As value at 0.99 quantile is of 214, so limiting it till 214 only and checking the shape again

```
In [18]: airbnb=airbnb[airbnb['number_of_reviews'] <=214]

In [19]: airbnb.shape

Out[19]: (47948, 16)
```

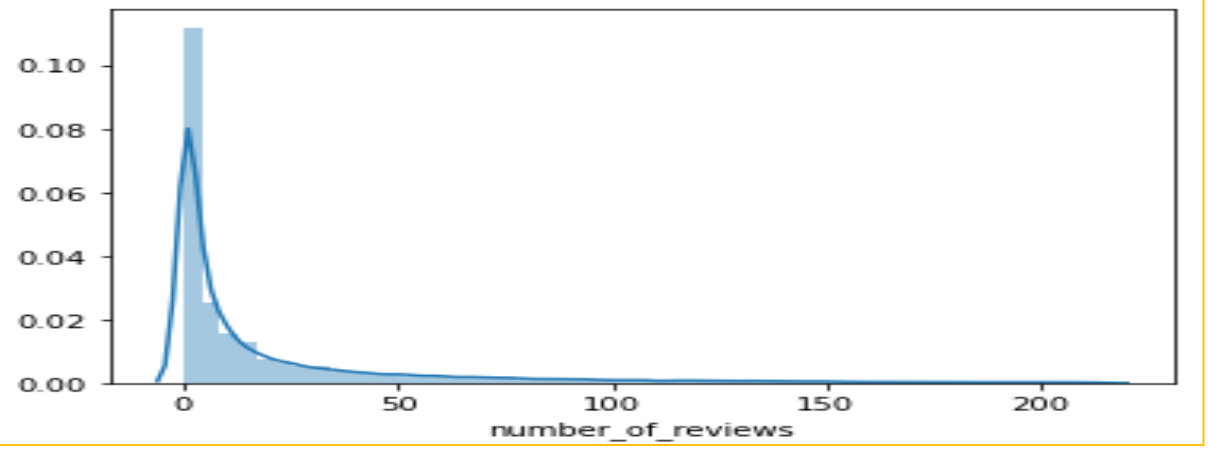
Now we checked the same by creating a boxplot again.

```
# Checking again outlier to see if it is fixed now
sns.boxplot(airbnb['number_of_reviews'])
plt.show()
```



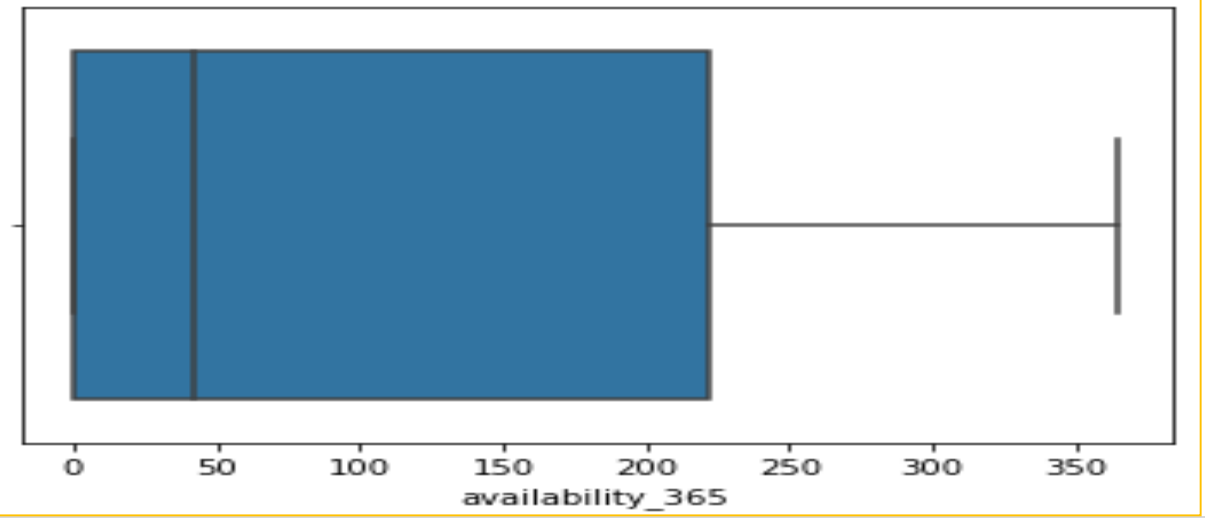
We confirmed it through a distribution plot too.

```
sns.distplot(airbnb['number_of_reviews'])
plt.show()
```



Now checking number_of_reviews. It is quite obvious that there is no outlier in availability_365 column.

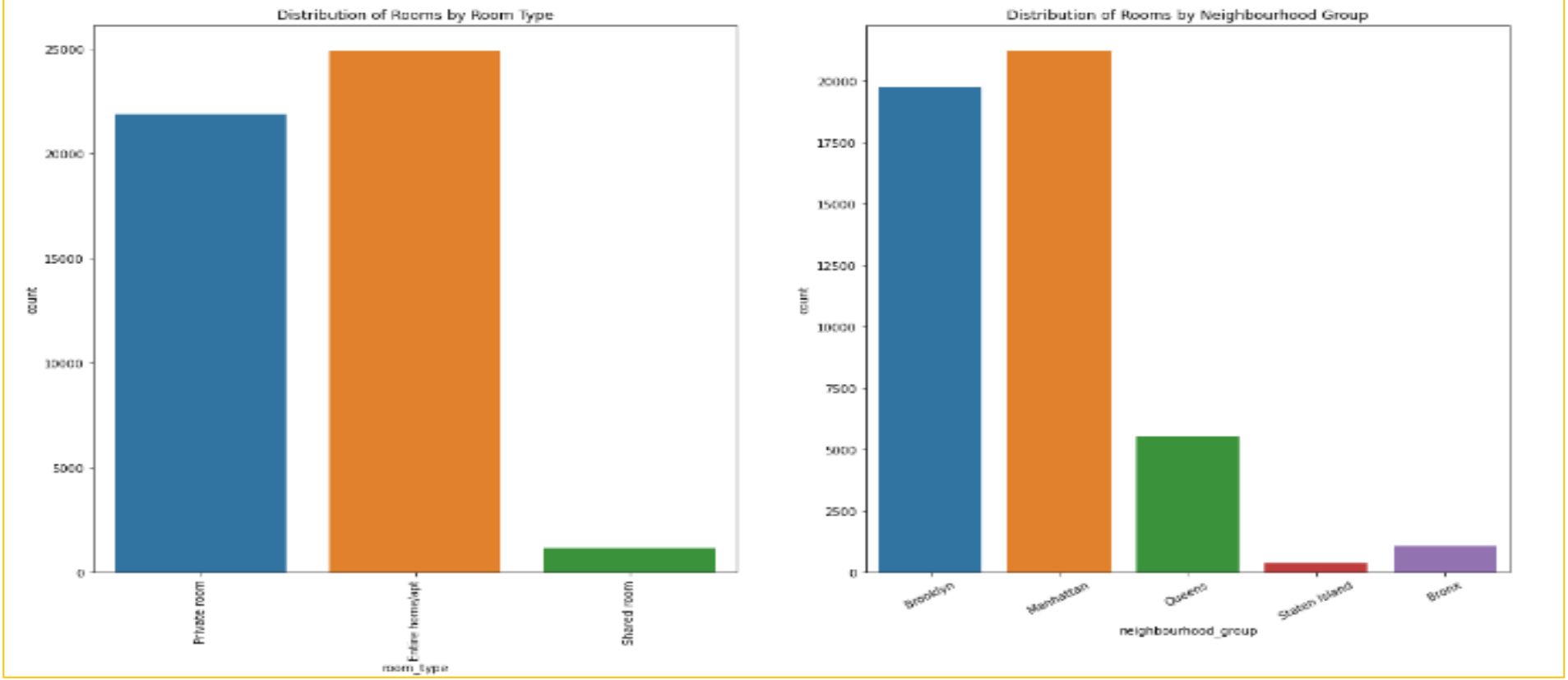
```
# Checking availability_365

sns.boxplot(airbnb['availability_365'])
plt.show()
```



EDA

Univariate Analysis on Categorical variables

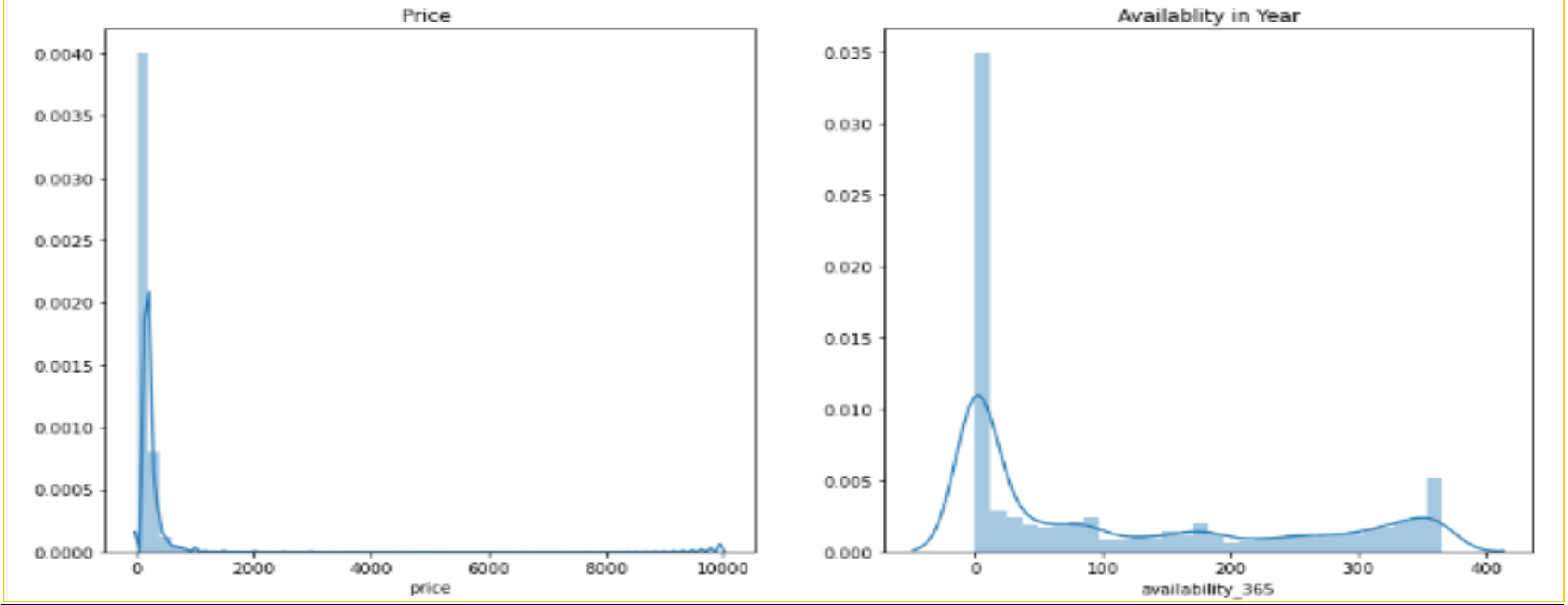
```
fig = plt.figure(figsize=(20,10))
plt.subplot(1,2, 1)
plt.title('Distribution of Rooms by Room Type')
sns.countplot(airbnb['room_type'])
plt.xticks(rotation=90)

plt.subplot(1,2, 2)
plt.title('Distribution of Rooms by Neighbourhood Group')
sns.countplot(airbnb['neighbourhood_group'])
plt.xticks(rotation=30)

plt.show()
```



Here, we have analysed room_type and neighbourhood_group and we can see that 'Entire home/apt' is more than other types in general and there are more rooms in 'Manhattan' than the other places.

Univariate Analysis on Categorical variables

```
fig = plt.figure(figsize=(15,7))
plt.subplot(1,2, 1)
plt.title('Price ')
sns.distplot(airbnb['price'])

plt.subplot(1,2, 2)
plt.title('Availablity in Year')
sns.distplot(airbnb['availability_365'])

plt.show()
```



Here, we have analysed room_type and neighbourhood_group and we can see that -:

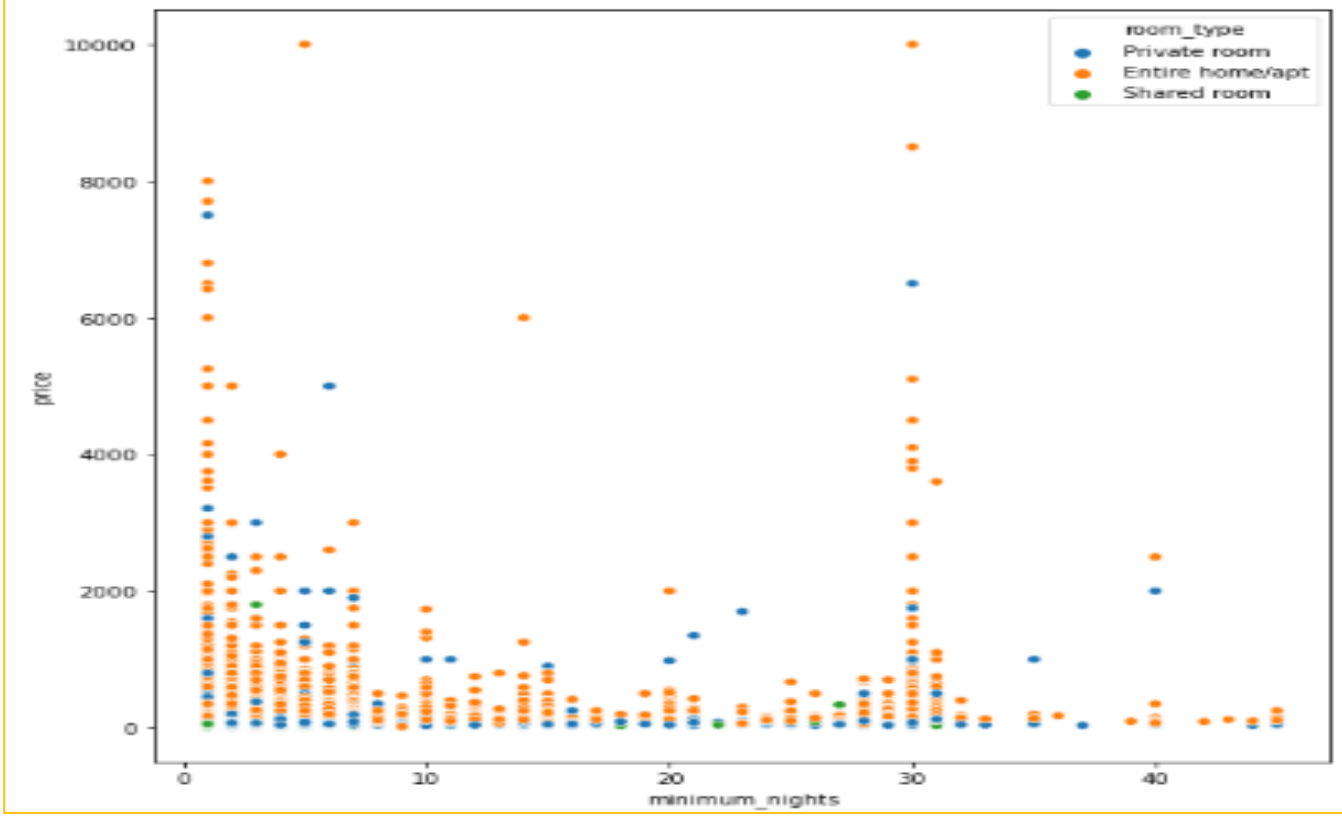
1. Most prices are less than 2000 but some are even close to 10000 showing some luxury places
2. Availability of room are mostly for 1 nights and some are little bit more than usual for 350 days

Bivariate Analysis

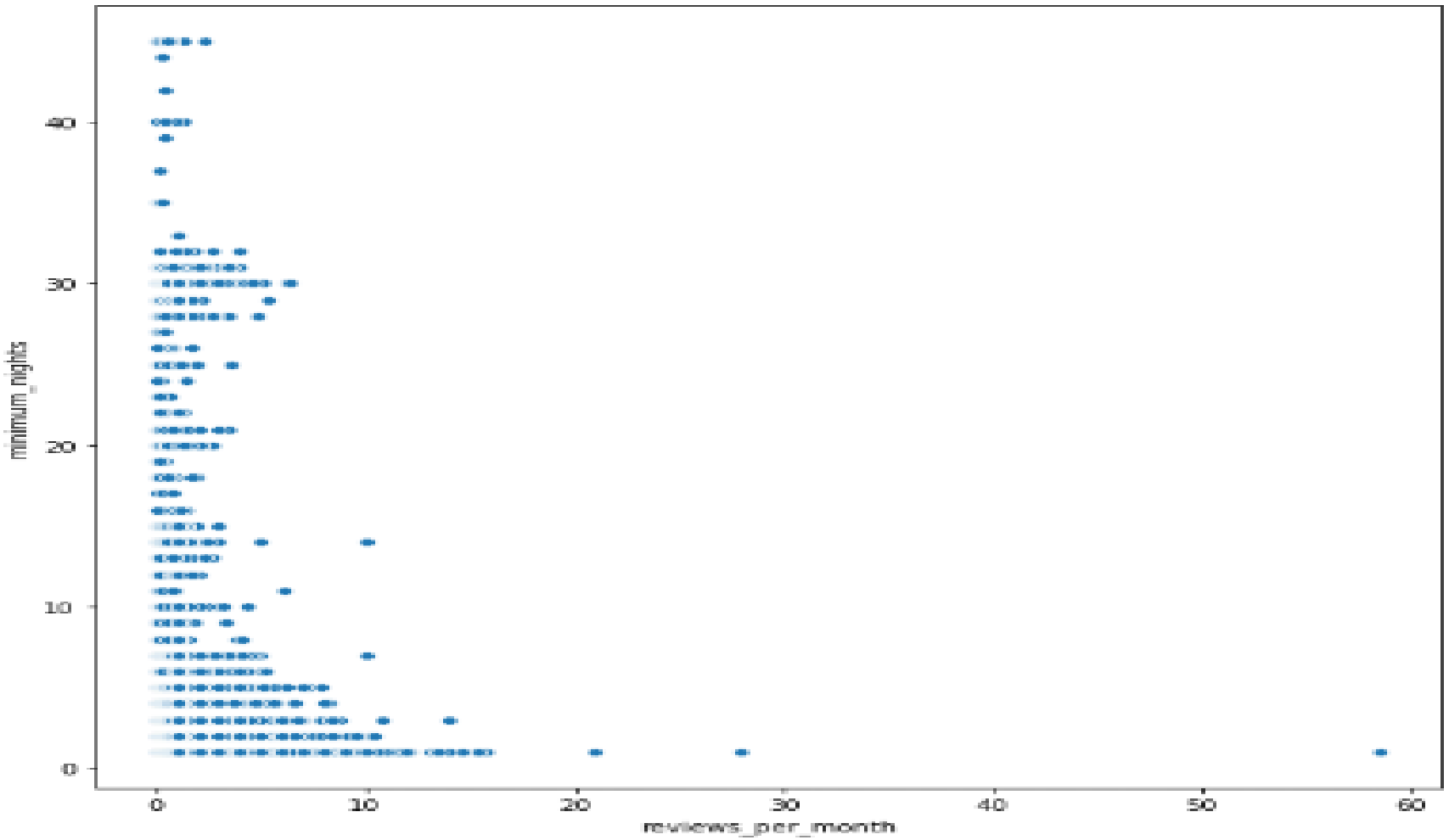
Here, we have tried to establish the relationship between room_type and minimum_nights. It is evident from the graph that price for minimum_nights is some cases are very high for 1,5,30 nights.

```
fig = plt.figure(figsize=(20,10))
plt.subplot(1,2, 1)
sns.scatterplot(y='price', x= 'minimum_nights', data=airbnb, hue='room_type')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1cc54fb25e0>
```

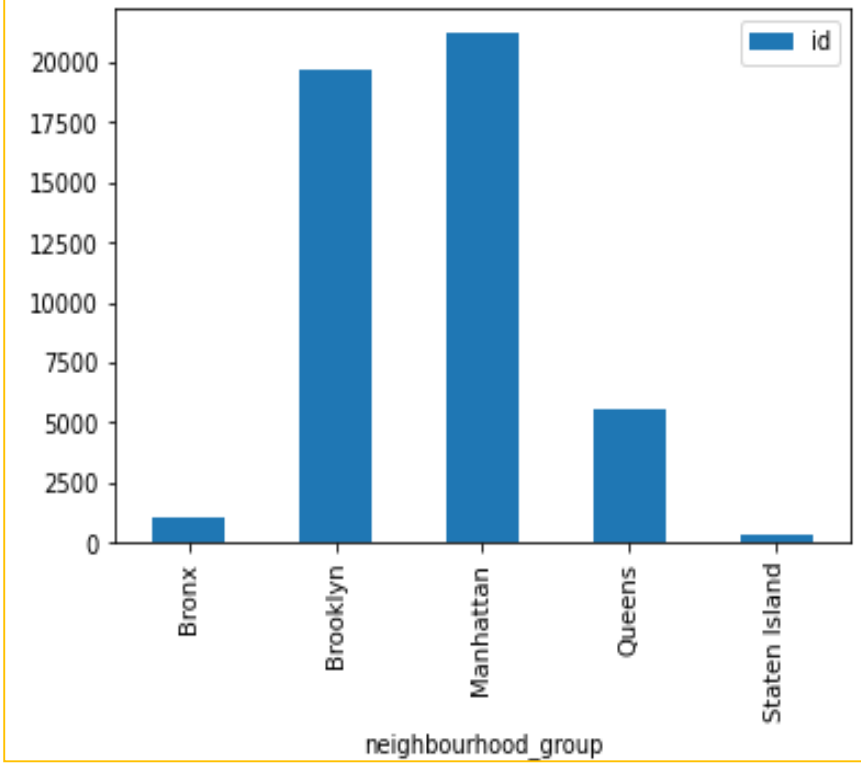


```
fig = plt.figure(figsize=(20,10))
plt.subplot(1,2,2)
sns.scatterplot(x='reviews_per_month', y= 'minimum_nights', data=airbnb)
plt.show()
```



Here, we have analysed room_type and neighbourhood_group and it is evident that number of review_per_month is very high for 1 nights as compared to more than 1 nights.

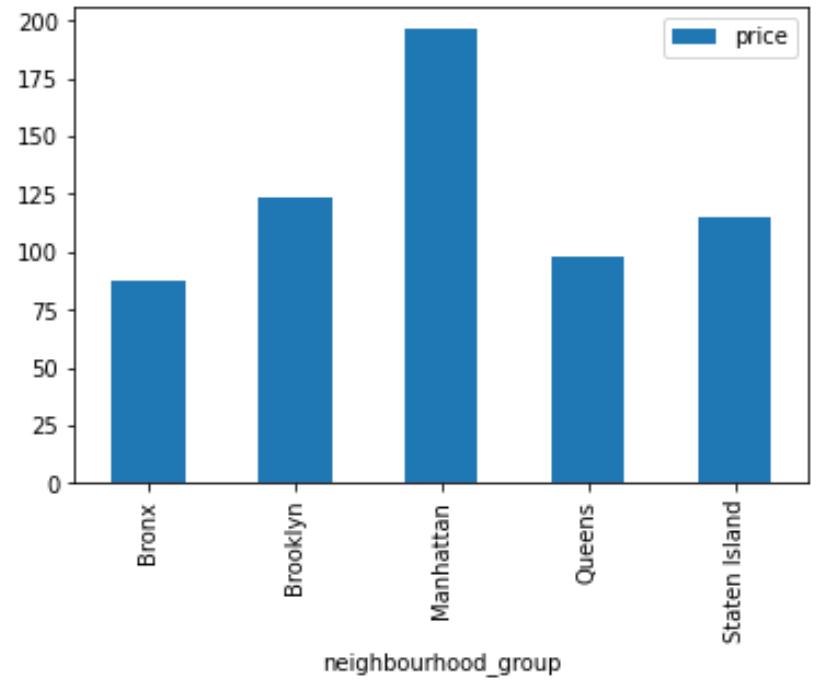
```
# Number of rooms by in New York:

listings_neighborhood_group = airbnb[['id','neighbourhood_group']].groupby('neighbourhood_group').count()
listings_neighborhood_group.plot.bar()
plt.show()
```



It is quite evident from the graph that-:

1. Brooklyn and Manhattan have the highest number of listings
2. Staten Island and Bronx have the least number of listings

```
# Average price of listings :

avg_price_listings =airbnb[['price','neighbourhood_group']].groupby('neighbourhood_group').mean()
px.bar(avg_price_listings)
avg_price_listings.plot.bar()
plt.show()
```

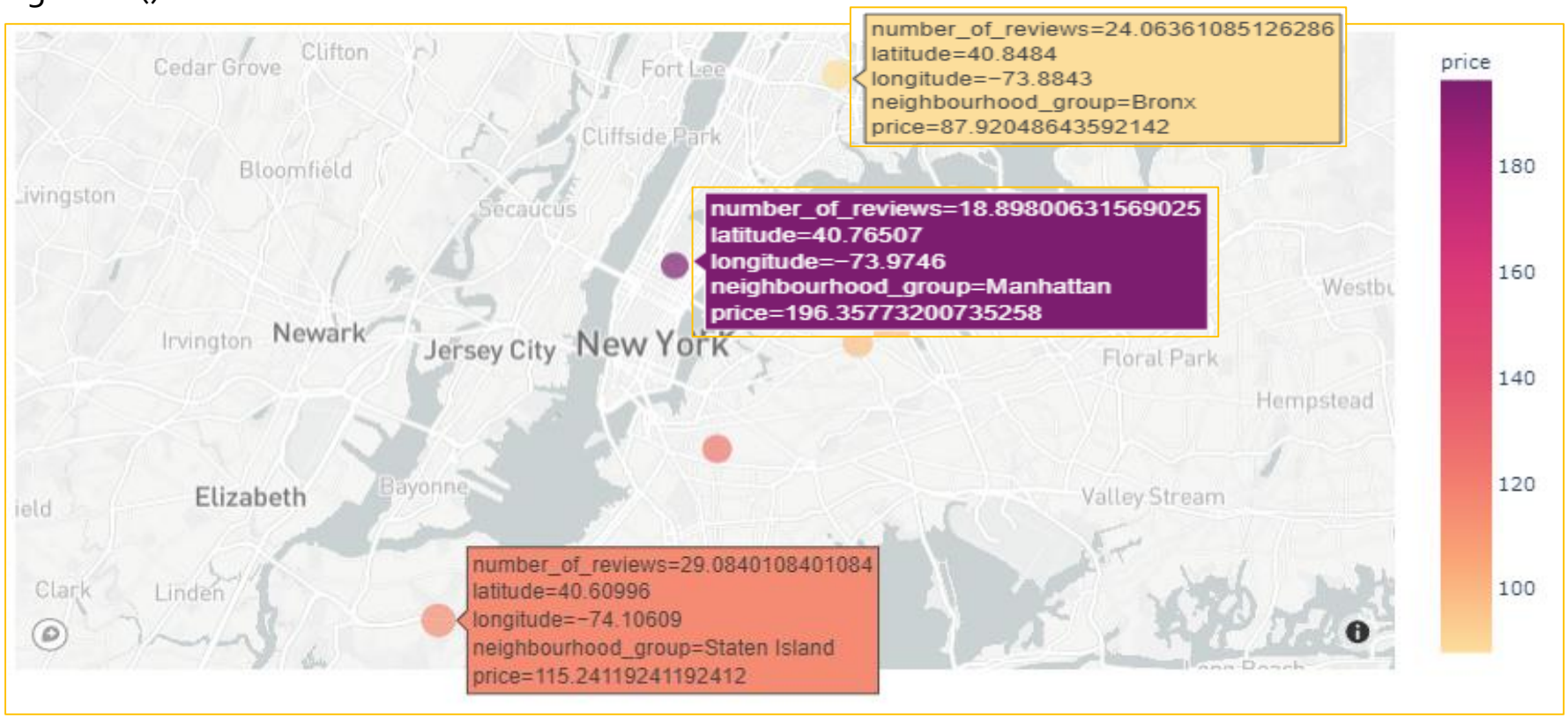


It is inferred from the graph that :-

1. Manhattan has the highest average price
2. Brooklyn is the next highest price
3. Bronx and Staten Island is little cheaper compared to the other cities

Distribution of price with reviews in New York:

```
borrows_grouped_data
=airbnb[['price','minimum_nights','number_of_reviews','neighbourhood_group','latitude','longitude']].groupby('neighbourhood_group').mean()
borrows_grouped_data = borrows_grouped_data.reset_index()
fig = px.scatter_mapbox(borrows_grouped_data, lat="latitude", lon="longitude", color="price",
size="number_of_reviews",color_continuous_scale='sunsetdark',hover_data=["neighbourhood_group"],
size_max=15, zoom=10)
fig.show()
```



We can infer from the graph that :-

1. Manhattan is the borrow with highest average price
2. Brooklyn and Staten Island is the borrows with next highest price
3. Queens and Bronx is little cheaper compared to the other borrows

A. Localities and properties in New York currently by number of available stays:

```
rtn = pd.DataFrame(airbnb.groupby(['room_type','neighbourhood_group'], as_index=False)['id'].count())

fig = px.bar(rtn, x='neighbourhood_group', y ='id',color ='room_type',
title ='Localities and Properties by Number of available stays')

fig.update_xaxes(title_text = "Neighbourhood",title_font=dict(size=18, family='Courier', color='crimson'))
fig.update_yaxes(title_text = "Available stays",title_font=dict(size=18, family='Courier', color='crimson'))
fig.update_layout(title_font_family="Times New Roman", title_font_color="red")
fig.show()
```

It is evident from the graph that :-

1. In Manhattan, more number of homes and apartments are available for stays
2. In Brooklyn, more number of private rooms are available for stays
3. Bronx and Staten Island have very less number of listings
4. Shared rooms are very less in number compared to the other two types of listings

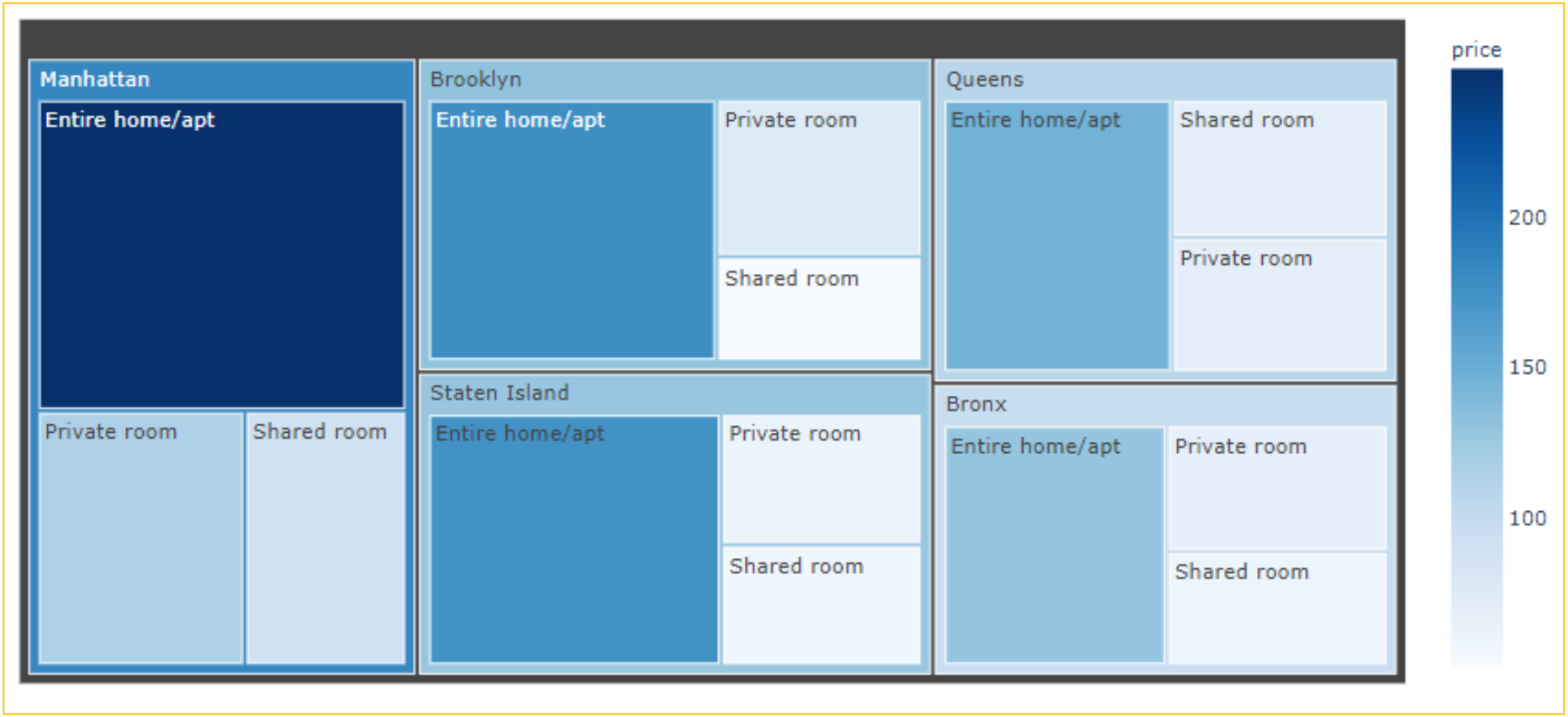
B. Price vs number of reviews :

```
fig1=px.scatter(airbnb,x='price',y='number_of_reviews')
fig1.show()
```

Most of the people prefer the listings that cost less than 2000 per night

C. Average price of listings per each type of room in each neighbourhood_group

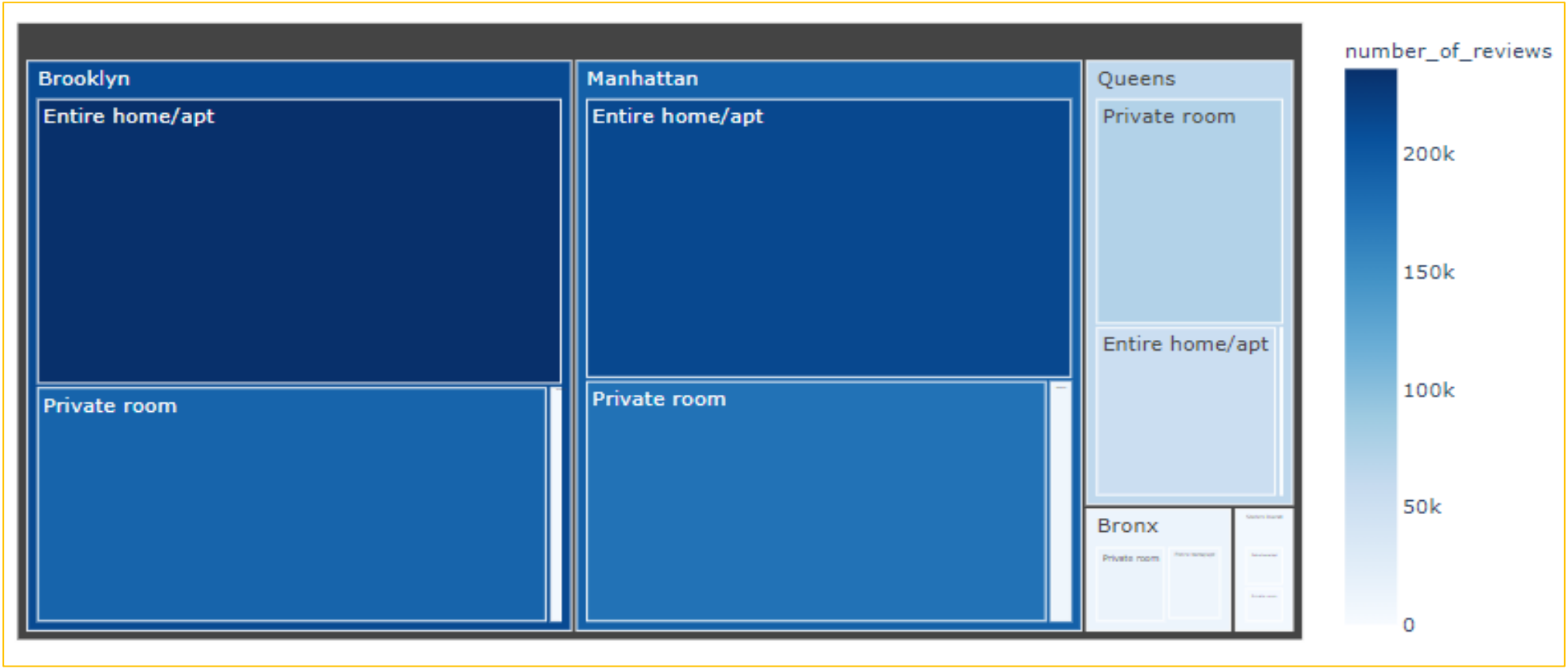
```
price_per_type_neighborhoodgroup =
airbnb[['room_type','neighbourhood_group','price']].groupby(['neighbourhood_group','room_type']).mean()

price_per_type_neighborhoodgroup = price_per_type_neighborhoodgroup.reset_index()

px.treemap(price_per_type_neighborhoodgroup,path=['neighbourhood_group','room_type'], values='price',
color='price',color_continuous_scale='blues')
```



1. In Manhattan, the average price of entire home/apt is higher than any other type of listings
2. Shared rooms and private rooms are costing lesser than renting an entire home/apt
3. In Queens, Staten island and Bronx, even the shared rooms are costing equal to private rooms

D. Most preferred areas and types of rooms

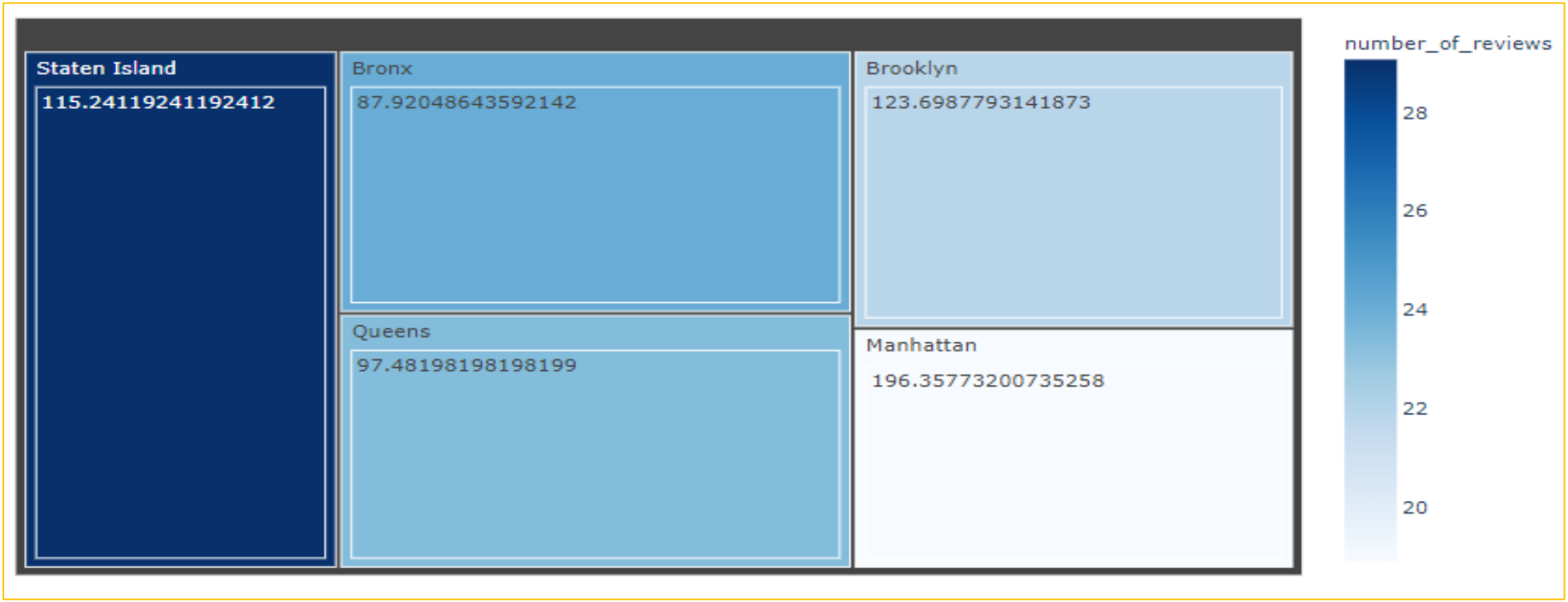
```
reviews_per_type_neighborhoodgroup
=airbnb[['room_type','neighbourhood_group','number_of_reviews']].groupby(['neighbourhood_group','room_type'])
.sum()

reviews_per_type_neighborhoodgroup = reviews_per_type_neighborhoodgroup.reset_index()

px.treemap(reviews_per_type_neighborhoodgroup,path=['neighbourhood_group','room_type'],
values='number_of_reviews', color='number_of_reviews',color_continuous_scale='blues')
```



1. In Manhattan and Brooklyn, entire home/apt stays are being mostly preferred
2. Private rooms are also being equally preferred as homes
3. In Queens, Bronx and Staten island, Private rooms are being preferred than homes/apts

E. Preferred area based on of price and reviews average

```
reviews_per_type_neighborhoodgroup =
airbnb[['price','neighbourhood_group','number_of_reviews']].groupby(['neighbourhood_group']).mean()

reviews_per_type_neighborhoodgroup = reviews_per_type_neighborhoodgroup.reset_index()

px.treemap(reviews_per_type_neighborhoodgroup,path=['neighbourhood_group','price'],
values='number_of_reviews', color='number_of_reviews',color_continuous_scale='blues')
```

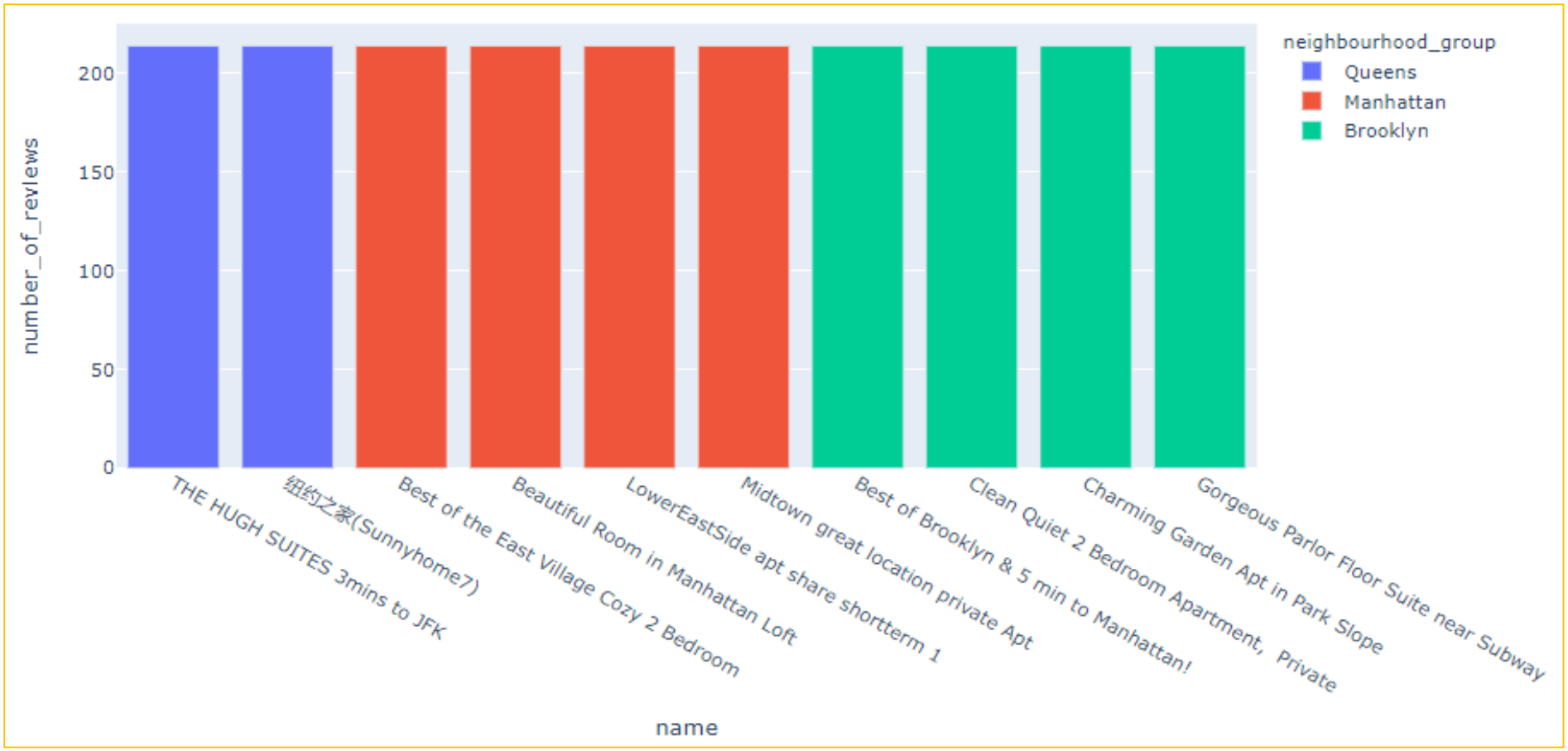


On the basis of average price and reviews, Manhattan is having the highest average followed by Brooklyn, Staten Island, Queens and Bronx

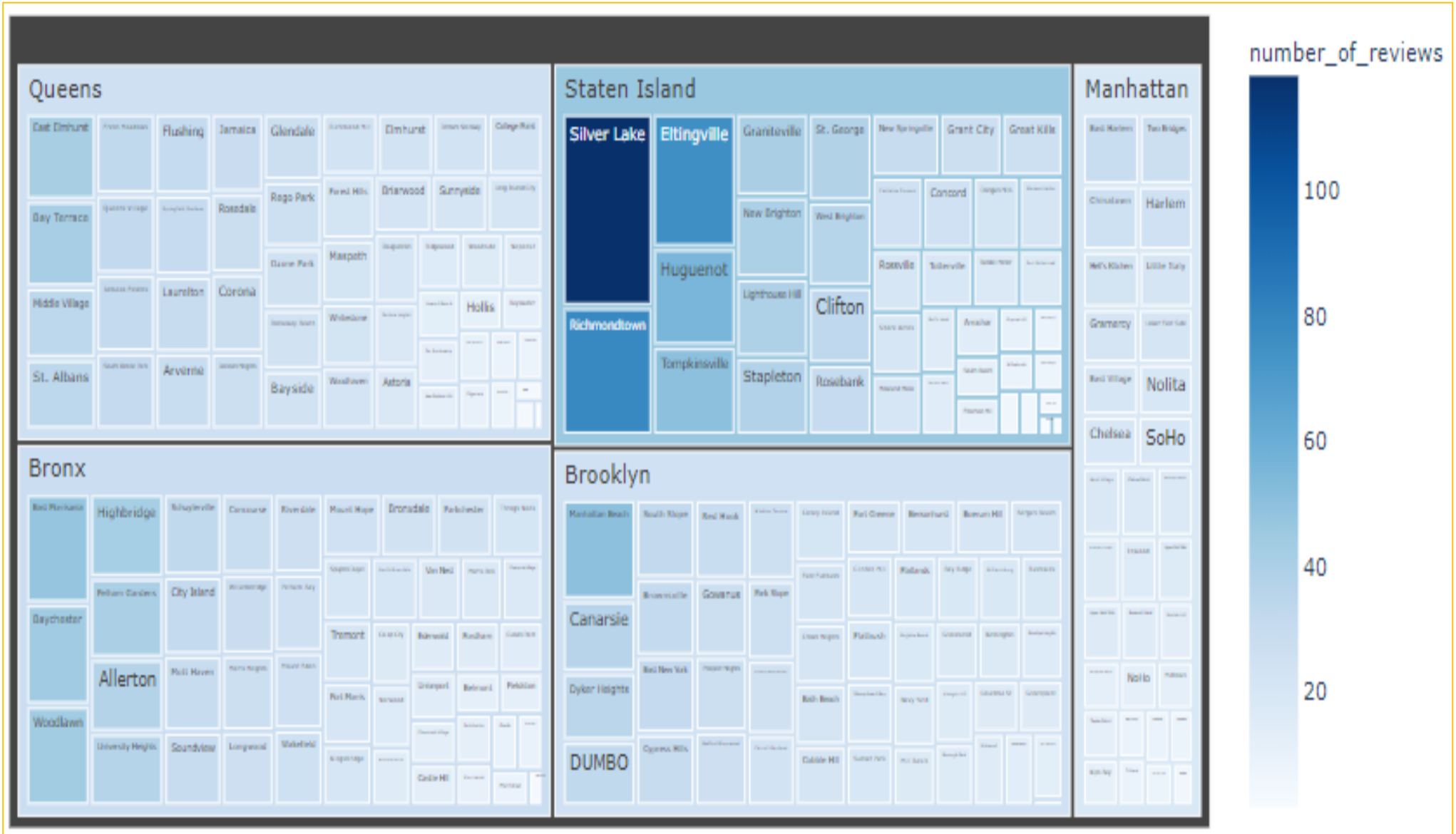
F. Top 10 preferred stays in Newyork city

top_10_preferred_stays =airbnb[['name','number_of_reviews','neighbourhood','neighbourhood_group']].sort_values('number_of_reviews',ascending=False).head(10)

px.bar(top_10_preferred_stays,x='name',y='number_of_reviews',hover_data=['neighbourhood'],color='neighbourhood_group')



G. Preferred neighborhoods in each borrow



The most preferred neighbourhoods in the entire Newyork city are

- Silver Lake (Staten Island)
- Richmondtown (Staten Island)
- Elting ville (Staten Island)
- East Elmhurst (Queens)
- Manhattan Beach (Brooklyn)

Limitation and Discussion

Limitations:

1. There are few column describing customer. Customer profile is not available so it is not possible to do full analysis on customer behavior and characteristics.
2. We assumed the data prior to the COVID – 19 period was achieving the desired revenue.
3. The company's strategies are decided considering the travel will increase in the post COVID period.
4. We assumed the company does not want to expand yet to new territories in NYC.
5. Since Customer profile is not available so it is not possible to do full analysis on customer behavior and characteristics.

Discussion:

Mathematically, there are outliers in price column but based on business domain knowledge we can assume that there might be luxury places for 10,000 per night. So, we didn't handle the outlier for price column.