Methodology

Step 1: **Data Cleaning**

- I have done most part of data cleaning in Python ipynb file and then exported the cleaned dataset into a csv file.

  Imputing Null Values:

- I checked for null values in the dataset. I aggregated the percentage of null values in each column.
  Code Snippet:

```
In [4]: df.isna().sum()/len(df)*100

Out[4]: id                                 0.000000
        name                               0.053175
        host_id                            0.000000
        host_name                          0.044994
        neighbourhood_group                0.000000
        neighbourhood                      0.000000
        latitude                           0.000000
        longitude                          0.000000
        room_type                          0.000000
        price                              0.000000
        minimum_nights                     0.000000
        number_of_reviews                  0.000000
        last_review                       20.558339
        reviews_per_month                 20.558339
        calculated_host_listings_count     0.000000
        availability_365                   0.000000
        dtype: float64
```

- Null values were not more than 20.5% in any of the column. Hence there was no need to drop any of the columns.
- I imputed the reviews_per_month column with the median and the last_review column with mode.
  Code Snippet:

```
df['reviews_per_month'].fillna(df.reviews_per_month.median(),inplace=True)
df['last_review'].fillna(df.last_review.mode()[0],inplace=True)
```

Handling Outliers:

- I used the describe function and plotted box plots for the numeric variables to check the spread of the data.

Code Snippets:

```
for x in num_vars:
    sns.boxplot(df[x])
    plt.show()
```

```
df.price.describe()
```

```
count    48421.000000
mean       137.543917
std        103.789003
min          0.000000
25%         69.000000
50%        105.000000
75%        175.000000
max        799.000000
Name: price, dtype: float64
```

- I found outliers in almost all the columns except variable availability_365.
  Code Snippet:

```
df[num_vars].describe()
```

	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_listings_count	availability_365
count	48421.000000	48426.000000	48415.000000	48409.000000	48895.000000	48895.000000
mean	137.543917	5.740181	20.707219	1.163021	7.143982	112.781327
std	103.789003	8.456492	35.810738	1.287604	32.952519	131.622289
min	0.000000	1.000000	0.000000	0.010000	1.000000	0.000000
25%	69.000000	1.000000	1.000000	0.270000	1.000000	0.000000
50%	105.000000	2.000000	5.000000	0.720000	1.000000	45.000000
75%	175.000000	5.000000	22.000000	1.520000	2.000000	227.000000
max	799.000000	45.000000	214.000000	6.800000	327.000000	365.000000

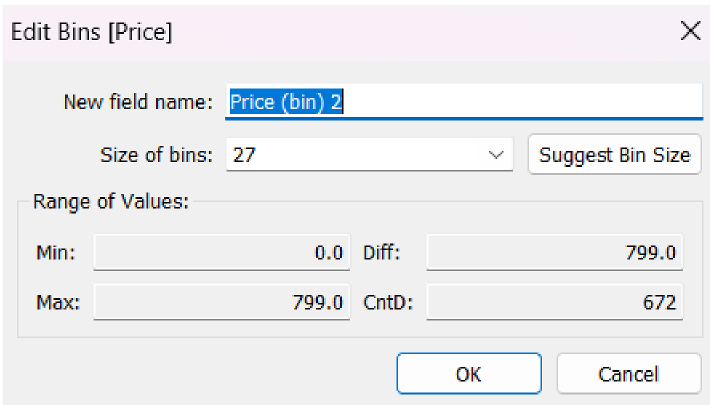
- I broke up the data into percentiles and found upto 99th percentile the data is gradually increasing but it spikes at the 100th percentile.
- I decided to cap all the variables at 99th percentile to prevent skewing of data apart from calculated_host_listings_count because mean of this variable was not required in the analysis.
  Snippet:

```
df.price = df.price[df.price<=df.price.quantile(0.99)]
df.minimum_nights = df.minimum_nights[df.minimum_nights <= df.minimum_nights.quantile(0.99)]
df.number_of_reviews = df.number_of_reviews[df.number_of_reviews <= df.number_of_reviews.quantile(0.99)]
df.reviews_per_month = df.reviews_per_month[df.reviews_per_month <= df.reviews_per_month.quantile(0.99)]
```

Step 2 : **Data Wrangling and Manipulation**

- I categorized the variables in 4 categories to smoothen the analysis. Categorical, Numeric, Location and Time Variables.

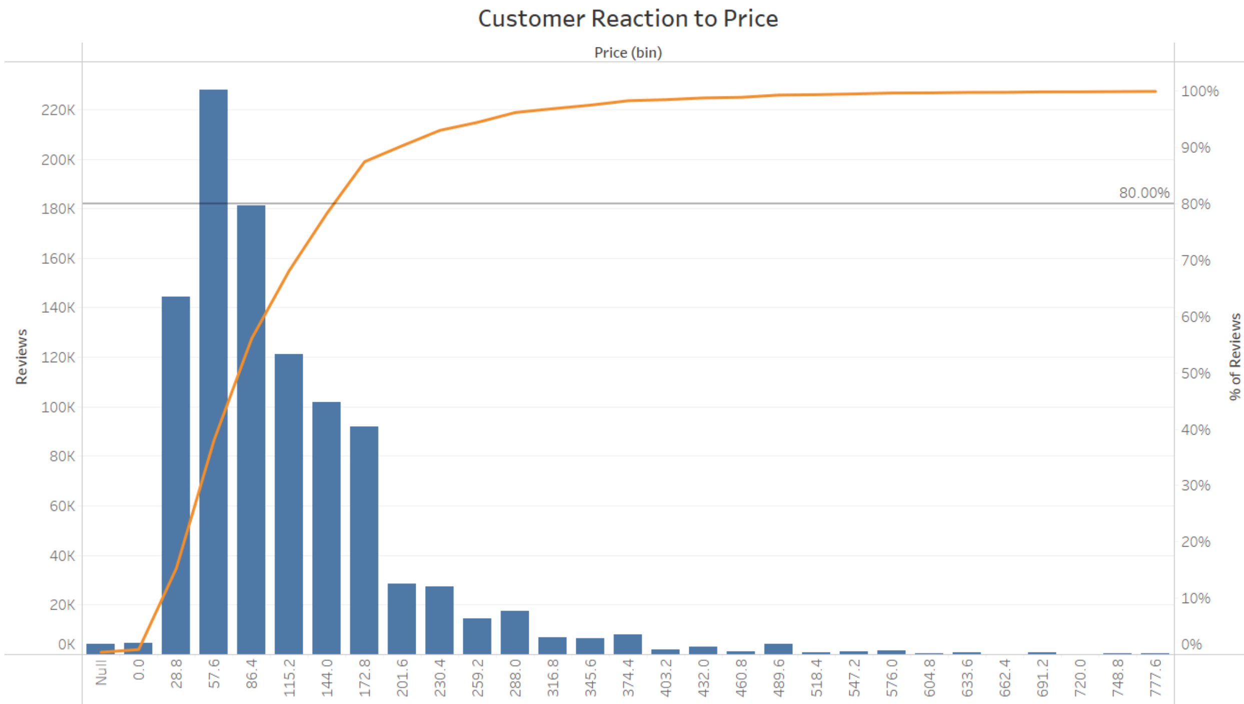
  Binning:
- I created bins for all the numeric variables and created separate groups or ranges. This helped me create a categorical measure out of numeric and compare them with other numeric measures.
  Snippet:



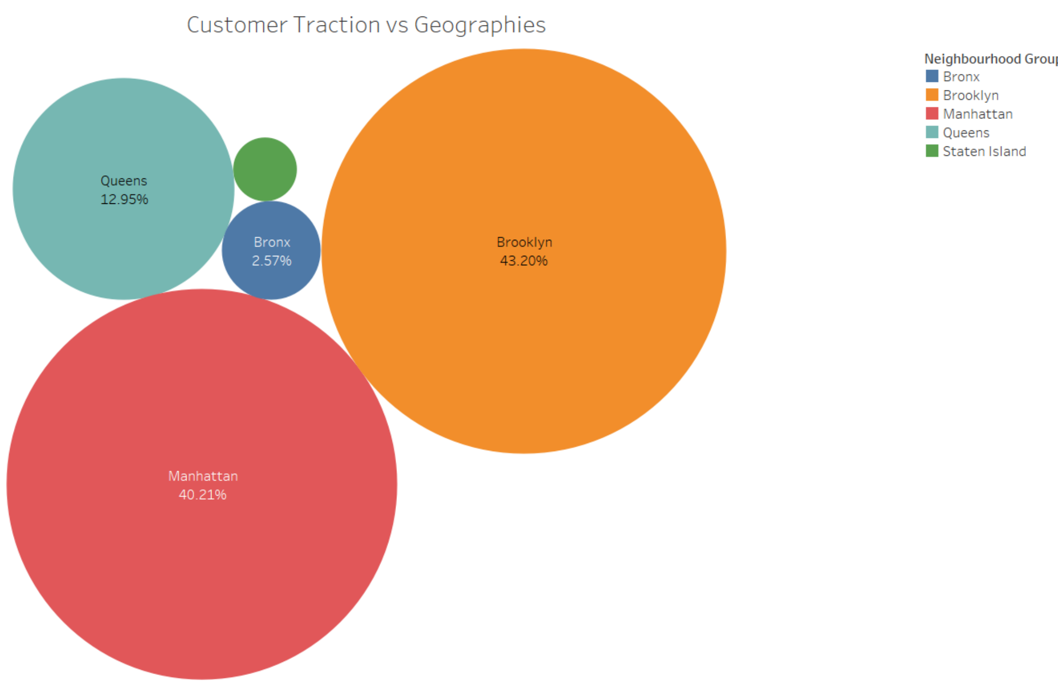
Step 3 : **Data Visualization and Analysis**

Presentation I:

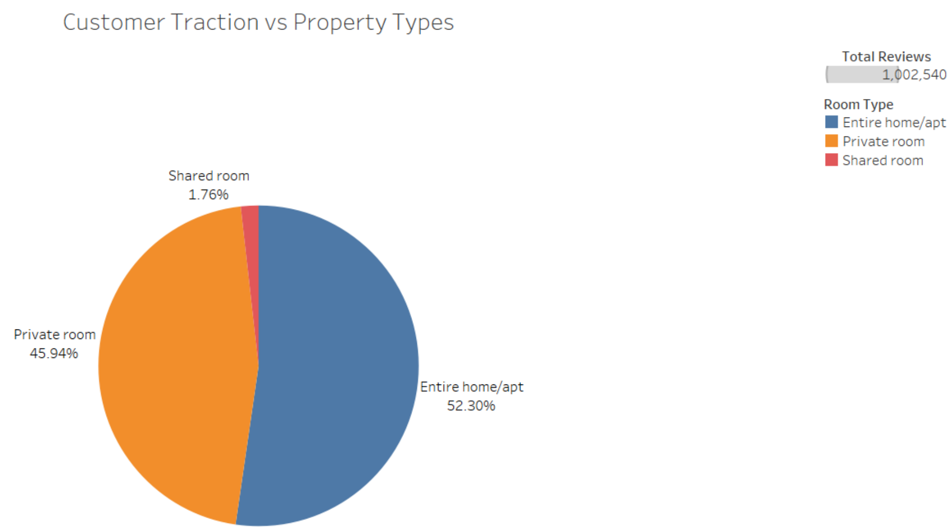
- I plotted a Pareto Chart for analyzing the traction among the price ranges. I used total reviews on y-axis and created bins in price variable on x-axis. I calculated the running total of the reviews and plotted a line that intercepts the 80% reference line. This helped to demonstrate where the 80% of the data lie.



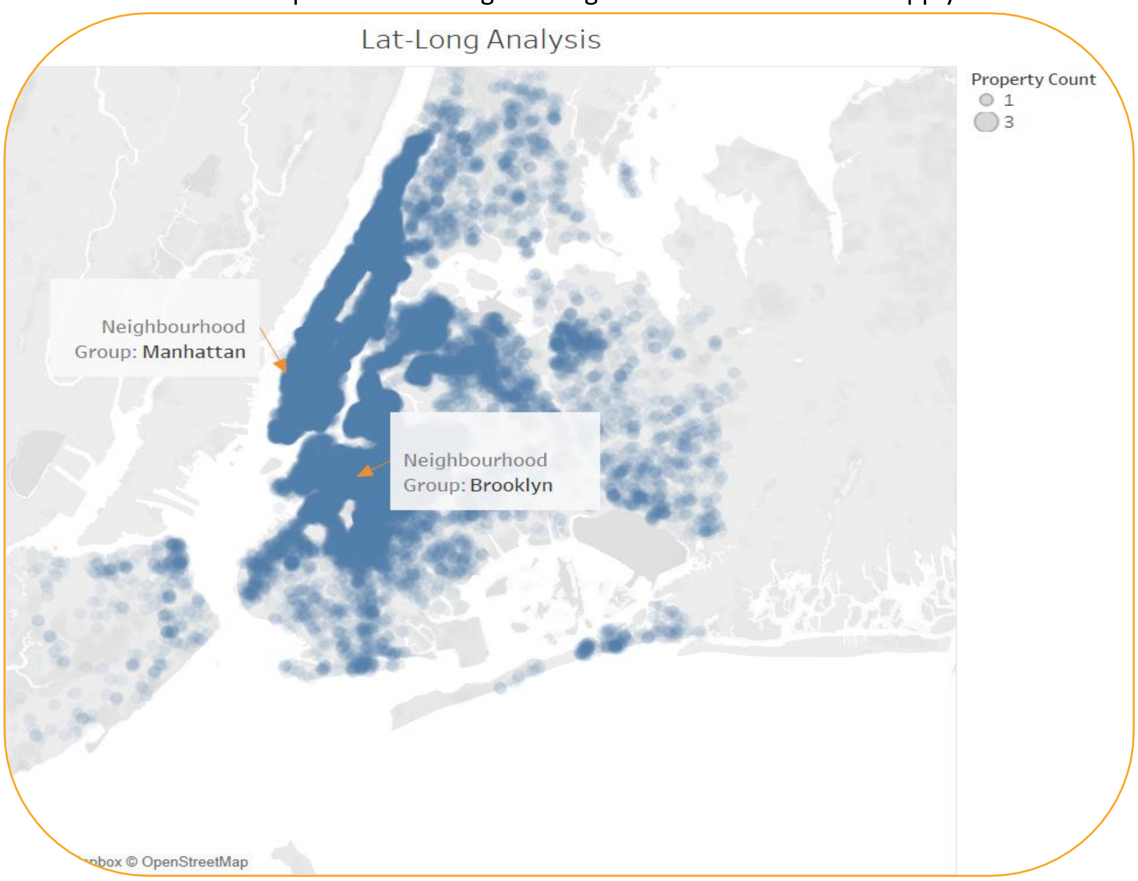
- To determine customer traction in terms of geographies I created a bubble chart with size of the bubble corresponding to the total reviews for the neighbourhood group and separate colors depicting the neighbourhoods.



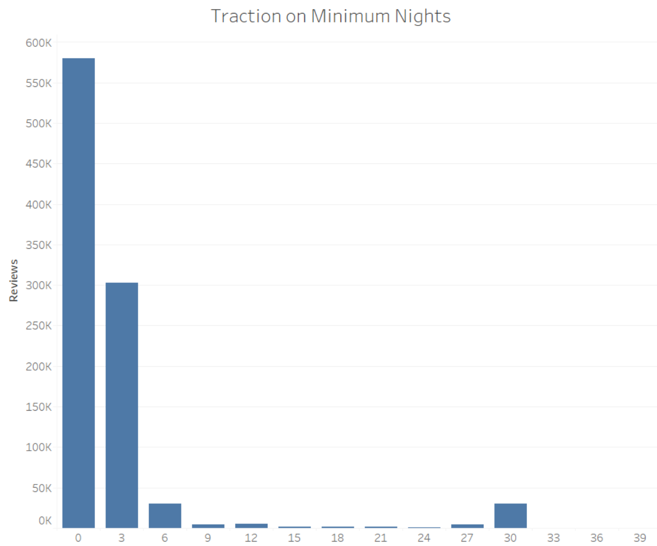
- For Property types I used a pie chart to showcase the proportion of total number of reviews attributed to each property type with each slice of the pie depicting each property type.



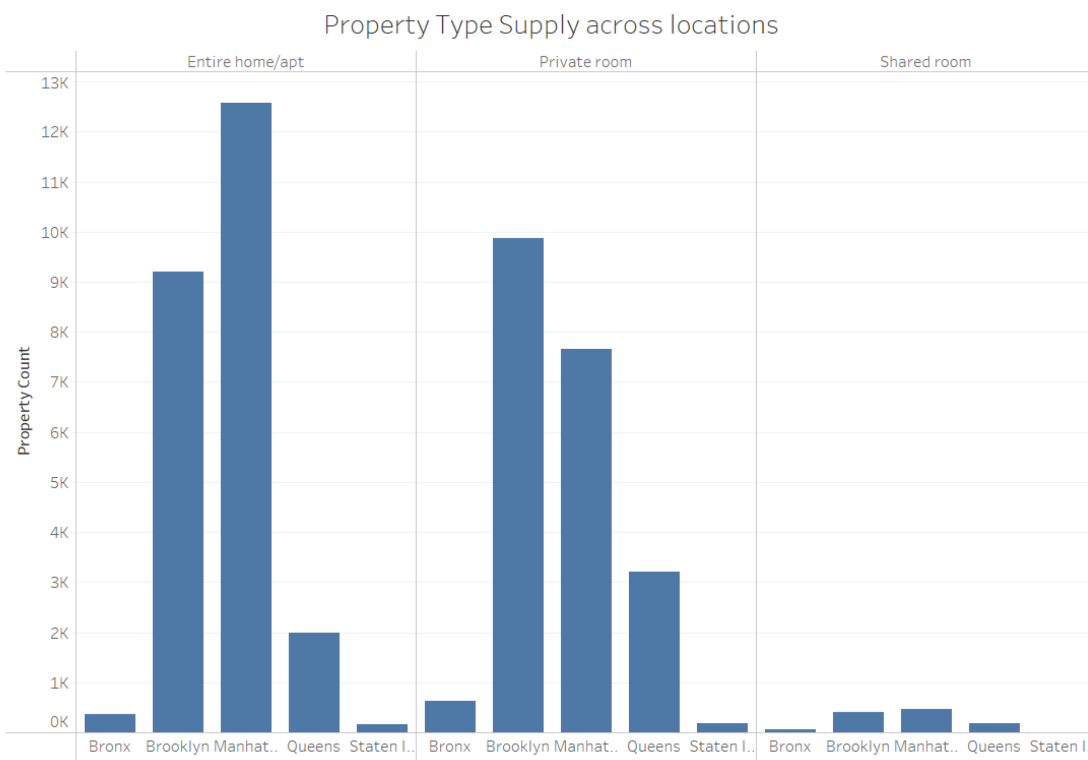
- To visualize the geographies with more customer traction I used latitude-longitude heat map with the spread of property listing location. I reduced the opacity of the color to showcase the clusters formed. This helped in visualizing the neighbourhoods with more supply and demand.



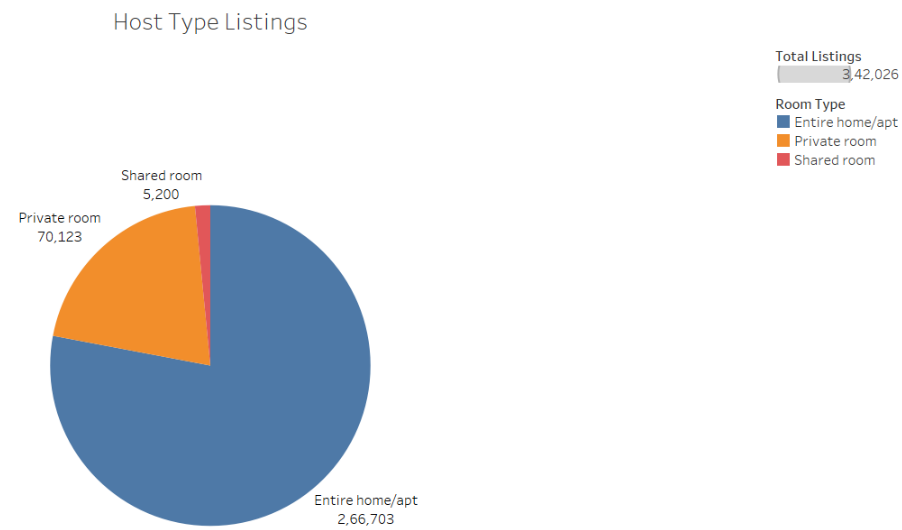
- In order to visualize customer traction with respect to some property features like the minimum nights feature, I grouped the minimum night variable into bins and plotted the number reviews against the ranges in a bar chart.



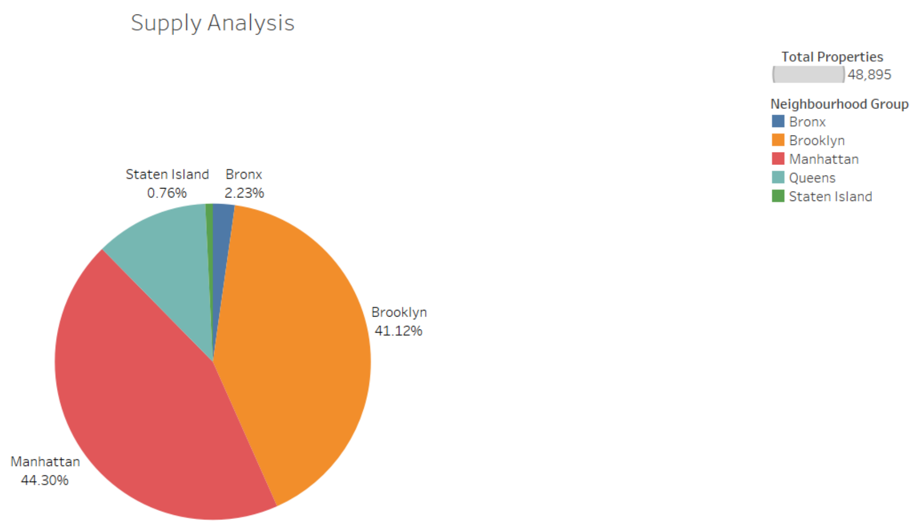
- To create visuals for the recommendation 1, I created a clustered column chart with the neighbourhood groups on x-axis and property count on y. I grouped the charts by property types to visualize the charts separately for separate property types.



- To create visuals for recommendation 2, I created a pie chart with each section depicting each host type and the labels as the number of listings collectively held by the host type.



- To visualize recommendation 3, I created another pie chart with each section corresponding to a neighbourhood group and the labels depicting the proportion of total property listings in percentages attributed to each neighbourhood.



Presentation II:

- I did a pivot grouped by various property types and average listings of hosts on the platform to determine the more active host type. I plotted a bar chart on the pivot.
  Snippet:

Row Labels	Average of calculated_host_listings_count
Entire home/apt	11
Private room	3
Shared room	5
Grand Total	**7**

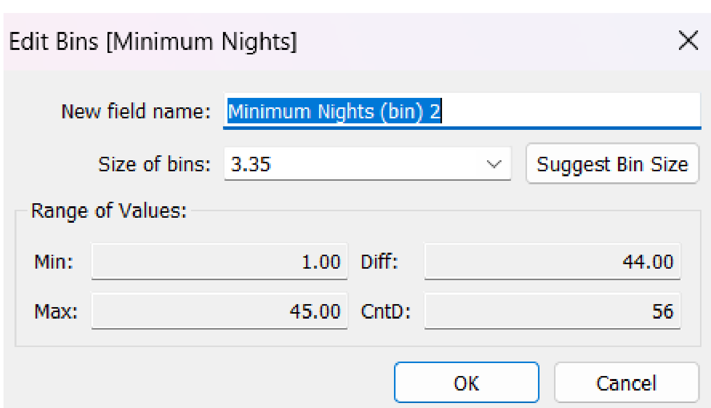
- I plotted stacked column chart of the total reviews against the neighbourhood groups and property types to determine the popular neighbourhoods and property types.

- I plotted the price bins created on x-axis as price ranges and reviews on the y-axis to understand the traction level of each price range among the customers. I used column chart to visualize.

- I plotted area chart to visualize the distribution of property types across various neighbourhoods. I also generated a pivot to look at the most popular localities among the neighbourhood groups.

- I visualized stacked column chart of total listings by different host types against the various neighbourhoods to determine which host type to target.

- I created bins of minimum nights variable to bifurcate it into ranges and plotted a bar chart with the total reviews on the y-axis to visualize customer traction among minimum nights requirement.



- I plotted a line chart of reviews using the last_review variable and visualized the trend with the last 9 years and respective quarters on the x-axis to analyze the demand trend.

- I created a pivot with a slicer to analyze the total properties, average price, total reviews and average reviews of different properties among different neighbourhood groups bifurcated into the localities inside the neighbourhoods. This granular visualization helped me to analyze how to increase traction among the unpopular properties and neighbourhoods.

Row Labels	Count of id	Average of price	Sum of number_of_reviews	Average of number_of_reviews2
Brooklyn	9559	165.39	267128	28
Manhattan	13199	216.94	235147	18
Queens	2096	140.81	60644	29
Bronx	379	125.20	11627	31
Staten Island	176	131.09	5857	33
Grand Total	25409	189.09	580403	23

room_type
Entire home/apt
Private room
Shared room

- The above visualization also helped to provide recommendations to the User experience team on how to manage the listings on the app and website.

Assumptions

- Variance - Data is evenly distributed or have equal variance across all categorical variables.
- Normal Distribution - Data is more or less normally distributed in price and other numeric variables.
- Independence - Data is independent of external factors like geopolitical factors.