Methodological document on Airbnb-NYC Data Analysis

Team: Gouri Phadtare, Chaitra Sanjee, Ramaya Ramchandran

Problem Statement: For the past few months, Airbnb has seen a major decline in revenue. Now that the restrictions have started lifting and people have started to travel more, Airbnb wants to make sure that it is fully prepared for this change.

Whom we are presenting:

1. Presentation – I :
   - Data Analysis Managers
   - Lead Data Analyst
2. Presentation – II:
   - Head of Acquisitions and Operations, NYC
   - Head of User Experience, NYC

Business Understanding:

Airbnb is an online platform that connects people who want to rent out their homes with people who want to stay in those homes.

How it works for hosts:

- Create a listing for your property
- Include a description, photos, and amenities
- Provide information about the local area
- Set your rates

How it works for guests :

- Create an account with a verified phone number and identification
- Search for listings using filters
- Select a property and make a reservation
- Pay online

How Airbnb makes money :

- Airbnb charges service fees to both hosts and guests
- Hosts typically pay a 3% service fee
- Guests typically pay a 6% to 15% service fee
- Airbnb may also collect and pay sales and tourism taxes

Now we can understood that from past few months, Airbnb has seen a major decline in revenue. Now that the restrictions have started lifting and people have started to travel more, Airbnb wants to make sure that it is fully prepared for this change.

Solution:

1.Data understanding:

Importing Data and important libraries and understanding data: Code snippets

```
# importing libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline

```

```
# ignore warnings
import warnings
warnings.filterwarnings('ignore')
pd.set_option('display.max_columns',None)
pd.set_option('display.max_rows',None)
```

```
# Loading the dataset
airbnb=pd.read_csv(r"C:\Users\Abvikas\Desktop\Case study Airbnb\AB_NYC_2019.csv")
airbnb.head()
```

```
# checking shapes, Dimension
airbnb.shape
```

```
(48895, 16)
```

Rows - 48895 Columns - 16

```
# columns
airbnb.columns
```

```
Index(['id', 'name', 'host_id', 'host_name', 'neighbourhood_group',
       'neighbourhood', 'latitude', 'longitude', 'room_type', 'price',
       'minimum_nights', 'number_of_reviews', 'last_review',
       'reviews_per_month', 'calculated_host_listings_count',
       'availability_365'],
      dtype='object')
```

Analysing column data types:

```python
# Datatypes of columns
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
```

Observations:

- Here we can understand that dataset has 48895 Rows and 16 Columns.
- We can categorize types of variables as follows from above:
    - **Numerical variables :**
      - price
      - minimum_nights
      - number_of_reviews
      - reviews_per_month
      - calculated_host_listings_count
      - availability_365
    - **Location Variables :**
      - latitude
      - longitude
    - **Time Variable:**
      - last_review
    - **Categorical Variable :**
      - id
      - name
      - host_id

- host_name
- neighbourhood_group
- neighbourhood
- room_type

Checking duplication in data by unique id : code snippets

```python
# finding unique values
airbnb.id.nunique()
```

48895

```python
# unique host_id
airbnb.host_id.nunique()
```

37457

```python
airbnb.describe()
```

host_id	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_listings_count	availability_365
4.889500e+04	48895.000000	48895.000000	48895.000000	48895.000000	48895.000000	38843.000000	48895.000000	48895.000000
3.762001e+07	40.728949	-73.952170	152.720687	7.029962	23.274466	1.373221	7.143982	112.781327
7.861097e+07	0.054530	0.046157	240.154170	20.510550	44.550582	1.680442	32.952519	131.622289
2.438000e+03	40.499790	-74.244420	0.000000	1.000000	0.000000	0.010000	1.000000	0.000000
7.822033e+06	40.690100	-73.983070	69.000000	1.000000	1.000000	0.190000	1.000000	0.000000
3.079382e+07	40.723070	-73.955680	106.000000	3.000000	5.000000	0.720000	1.000000	45.000000
1.074344e+08	40.763115	-73.936275	175.000000	5.000000	24.000000	2.020000	2.000000	227.000000
2.743213e+08	40.913060	-73.712990	10000.000000	1250.000000	629.000000	58.500000	327.000000	365.000000

Observations:

- We can see there is no duplication in listing data as size of unique id and row size is same .
- 37457 hosts are listed here
- From data description we can conclude :
    - It seems some entries with 0 price listing also some properties are costliest as max value is far apart from other quantiles
    - No. of reviews also started from 0 to max 629
    - Host listing count is maximum 327
    - Properties available are from 0 days to max 365

Checking null values present in data with percentage: code snippets

```
# finding null values:
airbnb.isnull().sum()/len(airbnb)*100
```

```
id                                 0.000000
name                               0.032723
host_id                            0.000000
host_name                          0.042949
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
```

Observations:

- Here we can understood that 'last_review' and 'reviews_per_month' columns have highest missing values that are 20.55 %
- 'name' and 'host name' column has 0.03 and 0.04 % missing values
- Here 'reviews_per_month' and 'last_review' column has missing purposely as there were no one to respond means they are not missing at random(MNAR). Hence people will not focus on these properties further.

Imputing Null values: Code snippets

```
# airbnb.reviews_per_month will impute with 0 as it has no reviews on for them
airbnb.fillna(0,inplace=True)

```

```
airbnb.reviews_per_month.isnull().sum() # succesfully imputed null values.
```

```
0
```

```
# name and host name column missing values is less we will impute that by 'unknown' as they are unknown

airbnb.name.fillna('unknown',inplace=True)
airbnb.host_name.fillna('unknown',inplace=True)
print(airbnb.name.isnull().sum())
print(airbnb.host_name.isnull().sum())
```

```
0
0
```

Assumptions for filling null values :

- Here reviews per month we filled as 0 as we assume no one has given review on that day hence we kept last review column as it is
- We assume name and host names are unknown as null values hence filled with 'unknown'

Extracting numerical Variables : We know 'id' , 'host id', 'longitude', 'lattitude' are categorical and location columns hence lets drop them from list

Numerical colums

```
num_col=airbnb.select_dtypes(include=['int64','float64']).columns
num_col=list(num_col)
print(num_col)
```

```
['id', 'host_id', 'latitude', 'longitude', 'price', 'minimum_nights', 'number_of_reviews', 'reviews_per_month', 'calculated_hos
t_listings_count', 'availability_365']
```

We know 'id' , 'host id', 'longitude', 'lattitude' are categorical and loacation columns hence lets drop them from list

```
num_col.remove('id')
num_col.remove('host_id')
num_col.remove('latitude')
num_col.remove('longitude')
num_col
```

```
['price',
 'minimum_nights',
 'number_of_reviews',
 'reviews_per_month',
 'calculated_host_listings_count',
 'availability_365']
```

2.Univariate Analysis :

a. Numerical variables

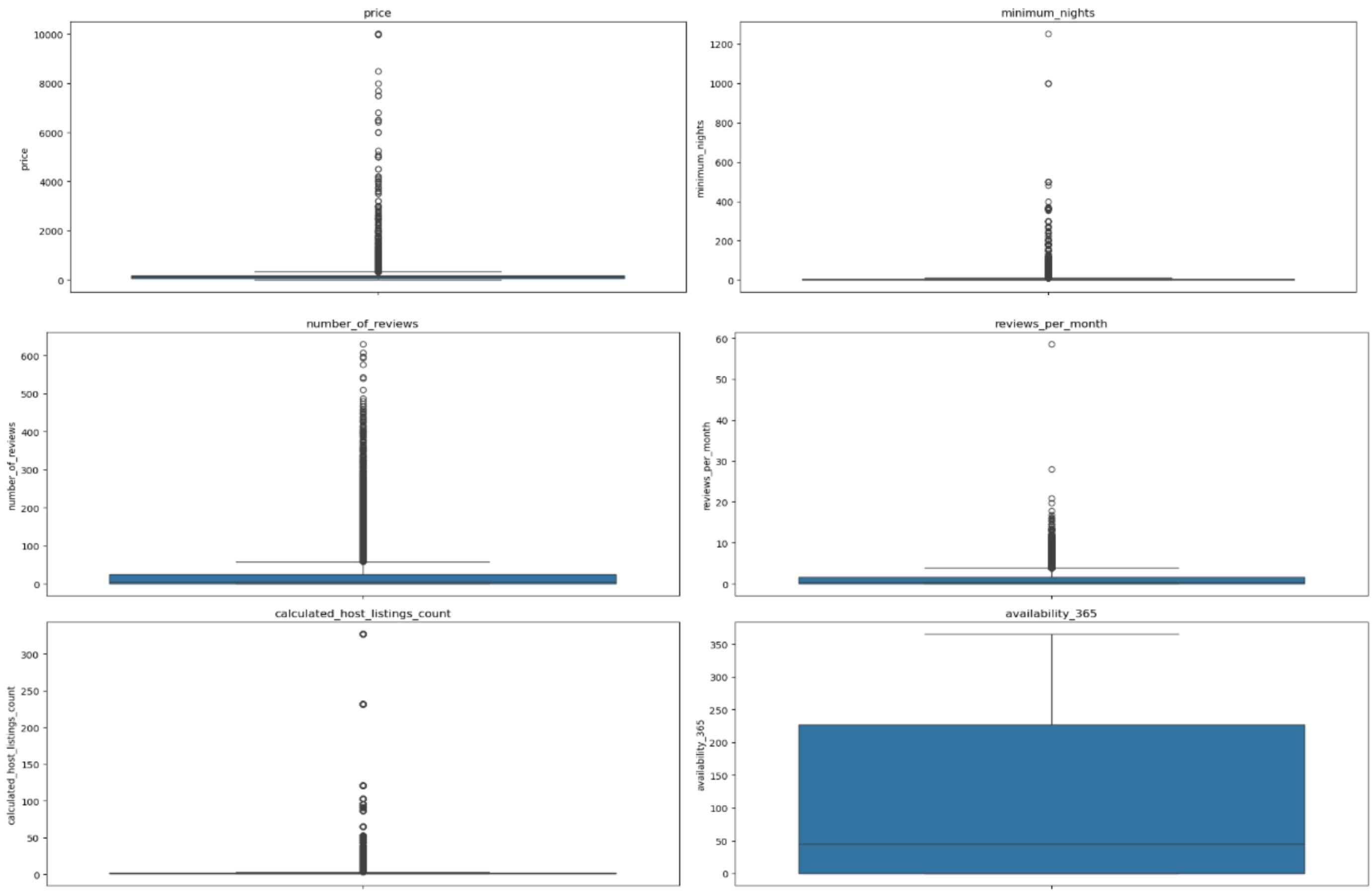
Plotting Box plot for numerical variables to check presence of outliers:

Box plots

```
# Plotting box plot
plt.figure(figsize=(20,22))

for n,col in enumerate(num_col):

    plt.subplot(5,2,n+1)    sns.boxplot(airbnb[col])
    plt.title(col)
    plt.tight_layout()
```



Observations:

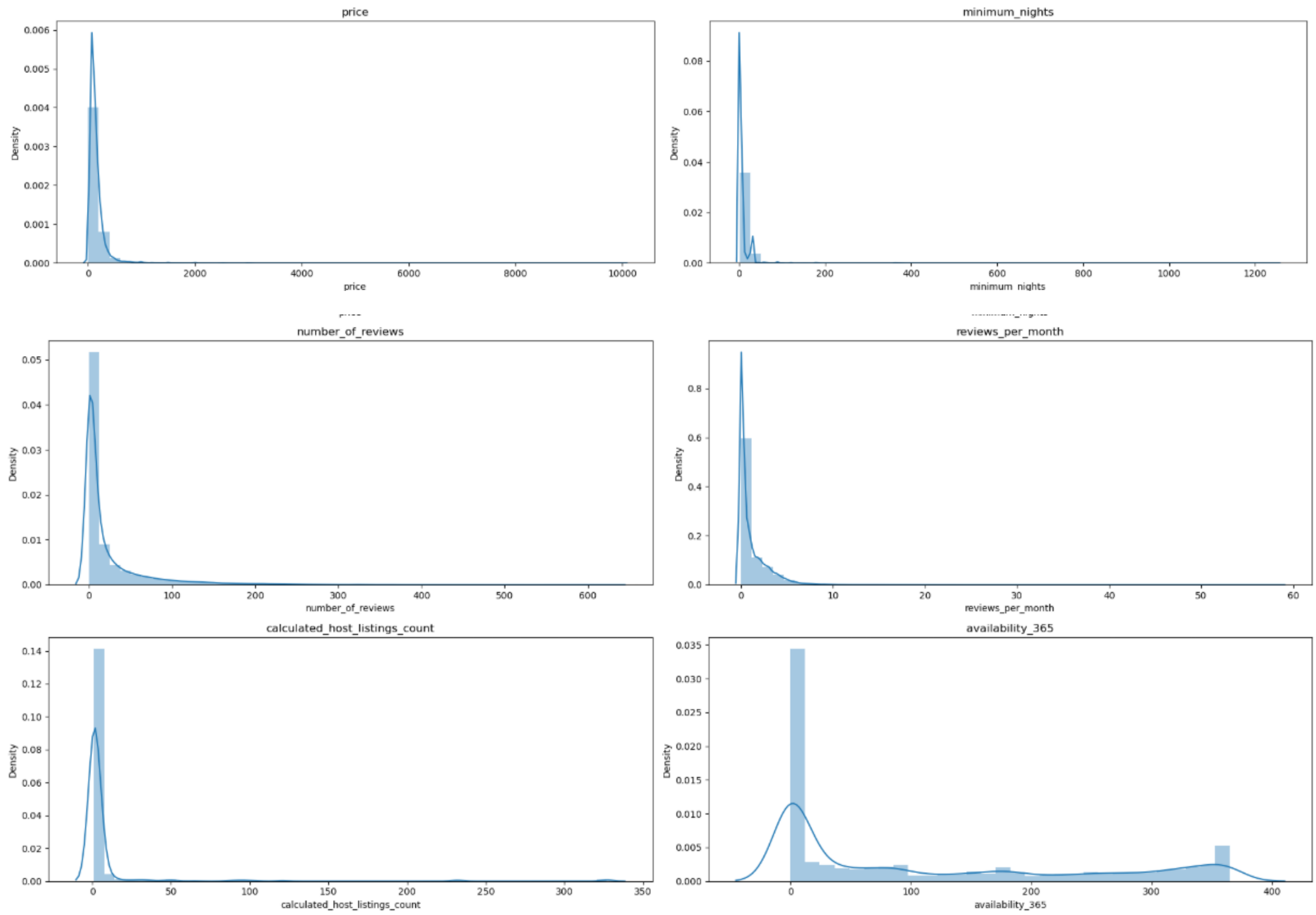
Looks like there are lots of outliers present in 'price', 'minimum_nights','number_of_reviews','reviews_per_month','calculated_host_listings_count'these variables.

Plotting Box plot for numerical variables to check presence of outliers:

```
# Plotting histograms for checking distribution over
plt.figure(figsize=(20,22))

for n,col in enumerate(num_col):

    plt.subplot(5,2,n+1)
    sns.distplot(airbnb[col])
    plt.title(col)
    plt.tight_layout()
```



Observations:

- Price -has normal distribution with right sided skew and spread over 0 to 1000
- Minimum Nights- has right skewed normal distribution over 0 to around 50
- Number of reviews -has right skewed normal distribution ranging from 0 to 200 with some spikes above
- reviews per month - has also right skewed distribution with spread of 0 to 10
- calculated_host_listings_count - has right skewed distribution from 0 to 100 and some spikes above

- Availability 365 - has normal distribution with long spread over right side up to 360

b.Categorical Variables :

Plotting count plot for checking count per category :

```
cat_var=airbnb.select_dtypes(include='object')
cat_var=list(cat_var)
cat_var
```

```
['name',
 'host_name',
 'neighbourhood_group',
 'neighbourhood',
 'room_type',
 'last_review']
```

Here name, host names are distinct entries also last_revievs is contains date so we will drop them.

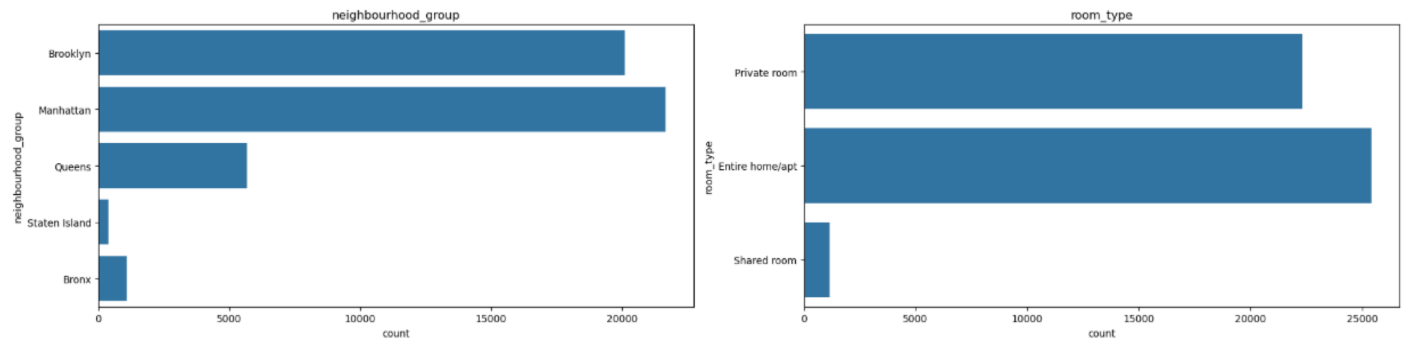
```
cat_var=['neighbourhood_group', 'room_type']
cat_var
```

```
['neighbourhood_group', 'room_type']
```

```
# Plotting histograms for checking distribution over
plt.figure(figsize=(20,22))

for n,col in enumerate(cat_var):

    plt.subplot(5,2,n+1)
    sns.countplot(airbnb[col])
    plt.title(col)
    plt.tight_layout()
```




```
airbnb.neighbourhood_group.value_counts()
```

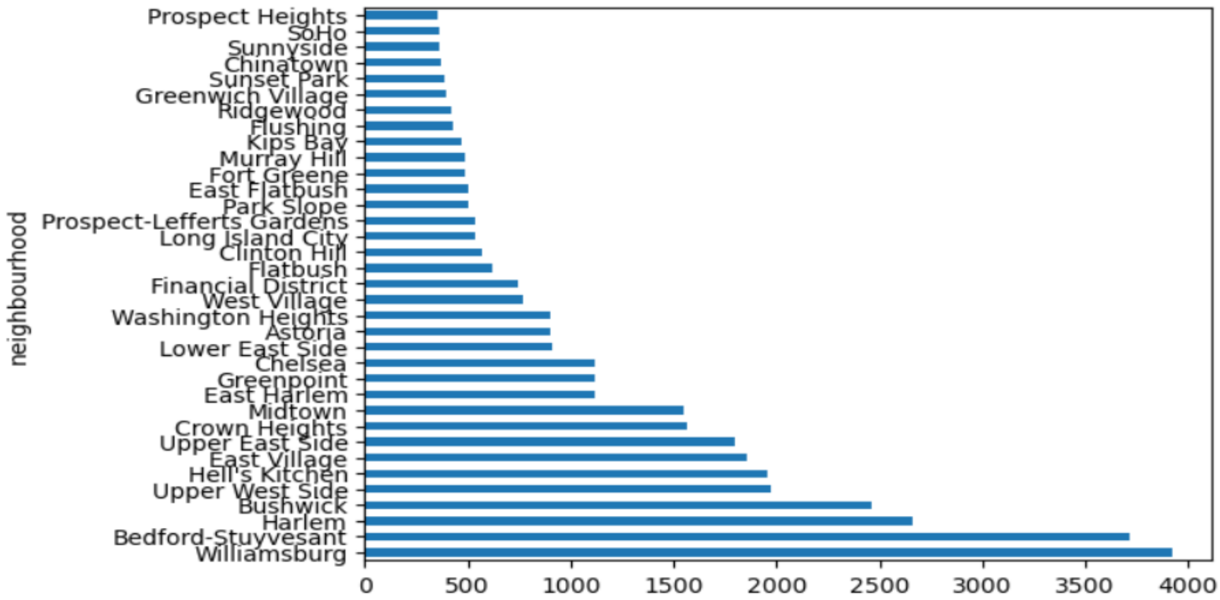
```
neighbourhood_group
Manhattan        21661
Brooklyn         20104
Queens            5666
Bronx             1091
Staten Island      373
Name: count, dtype: int64
```

```
airbnb.room_type.value_counts()
```

```
room_type
Entire home/apt    25409
Private room       22326
Shared room         1160
Name: count, dtype: int64
```

```
1  airbnb.neighbourhood.value_counts()[:35].plot.barh()
```

```
<Axes: ylabel='neighbourhood'>
```



Observation:

- **neighbourhood_group :**
  - Manhattan has 21661 listings are maximum than all cities
  - Brooklyn has 20104 listings
  - Queens has 5666 listings
  - Bronx has 1091 listings
  - Staten Island has 373 listings

- **room_type :**
  - Entire home/apt are maximum listed 25409 overall properties.
  - Private room has 22326 listings followed
  - Shared room has 1160 listings.

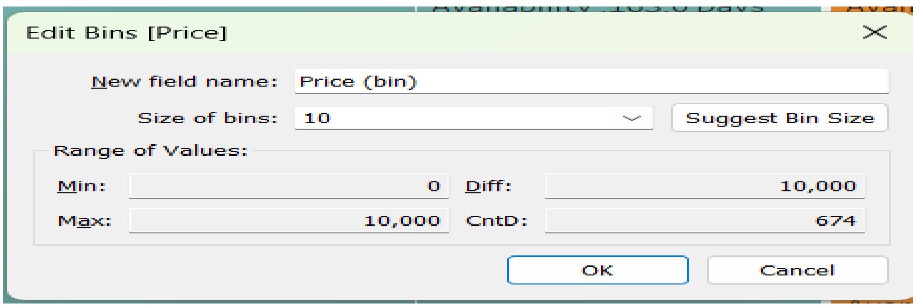
- **Neighbourhood:** Williamsburg has maximum listing present than others

3.Bivariate Analysis:

We have used tableau for analysis.

1.Customer Preferences:

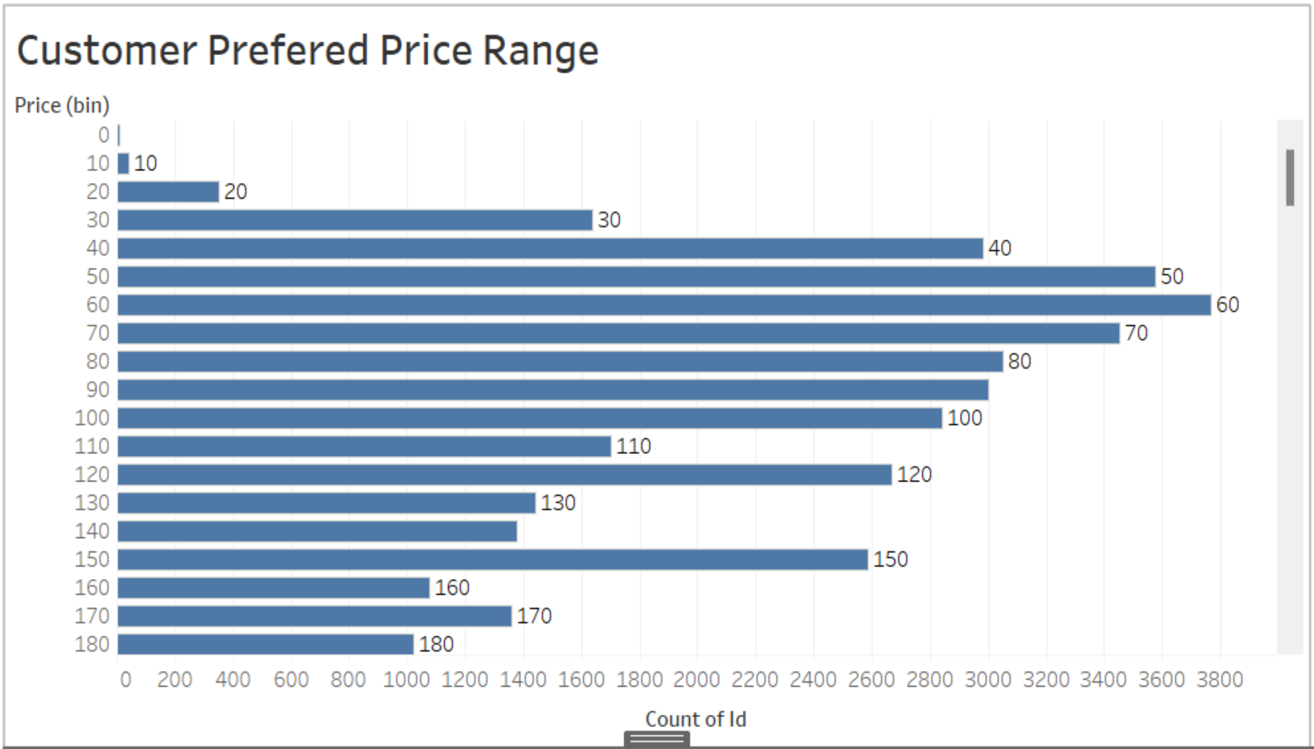
We have created price bins of size 10 to analyse price preferred.

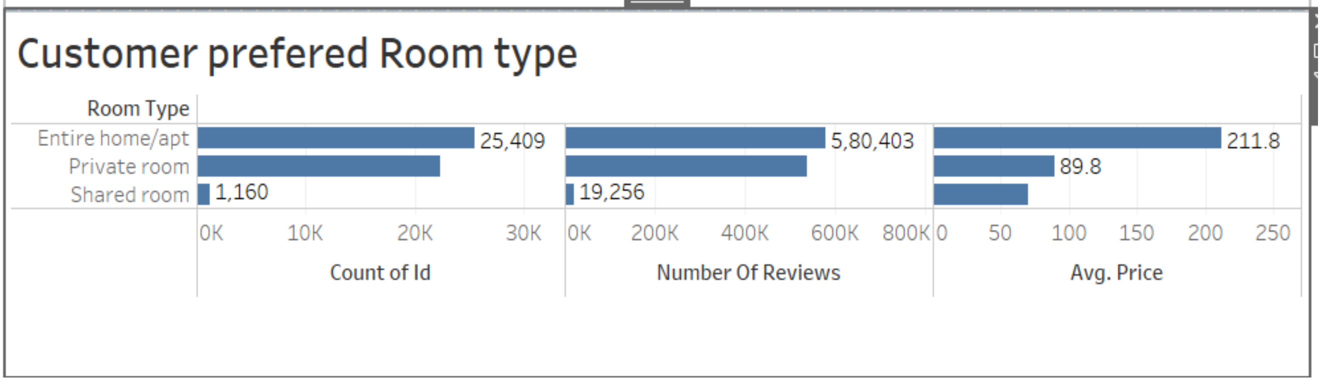


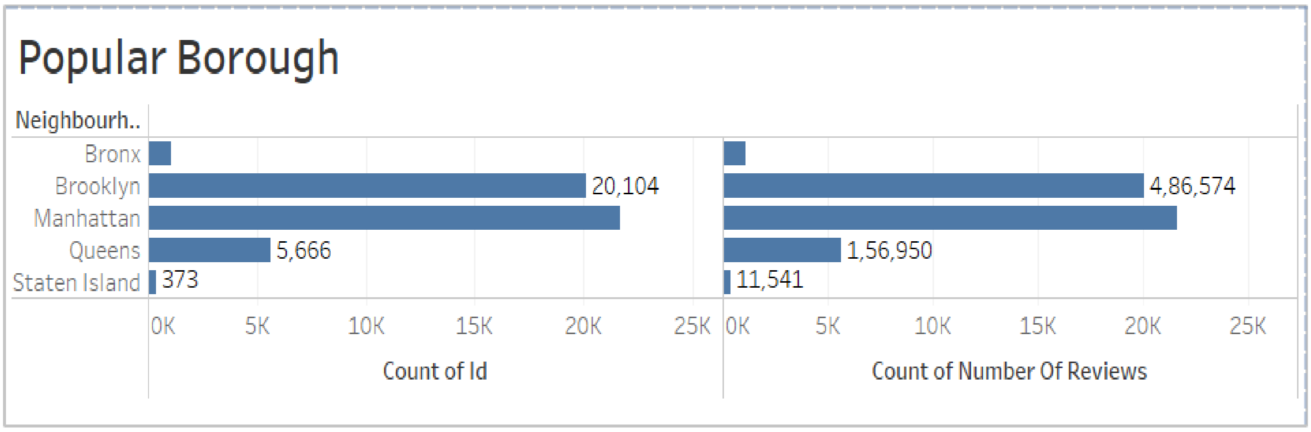
Observations:

- Low prices ranging from **60-150** preferred by customers
- **Entire home/Apt**. highly preferred by customers on basis of **price, number of reviews and listings**. Followed by **Pvt rooms**.
- **Manhattan** being most popular **borough** followed by **Brooklyn** for highest bookings and number of reviews.

Graphs:





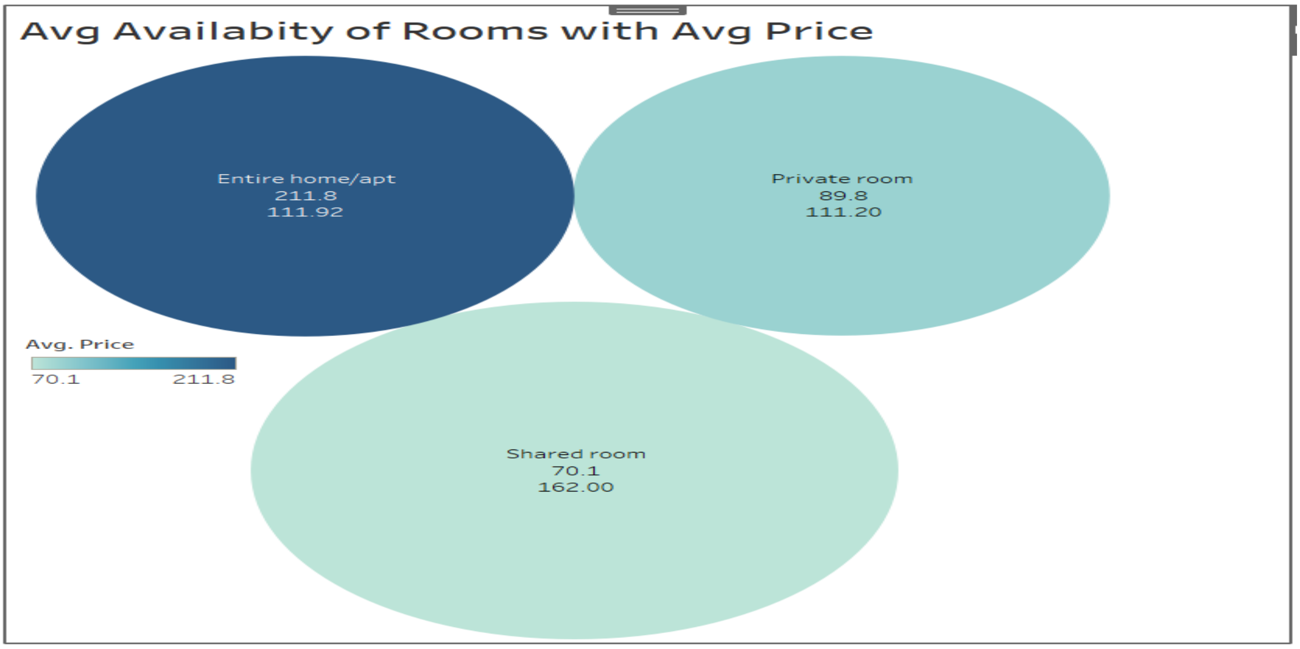


2.Availability of Rooms with Average Price:

Observations:

- Having a high price range with **average price 211$**, **Entire home/apt** types of rooms are available for **112 days**.
- **Private rooms** available for average of **111 days** with low average **price 90$**
- **Shared rooms** around **162 days** on average being available with the lowest in average **price 70$**

Graph:



3. Top 5 Hosts based on Price, Reviews and Listings:

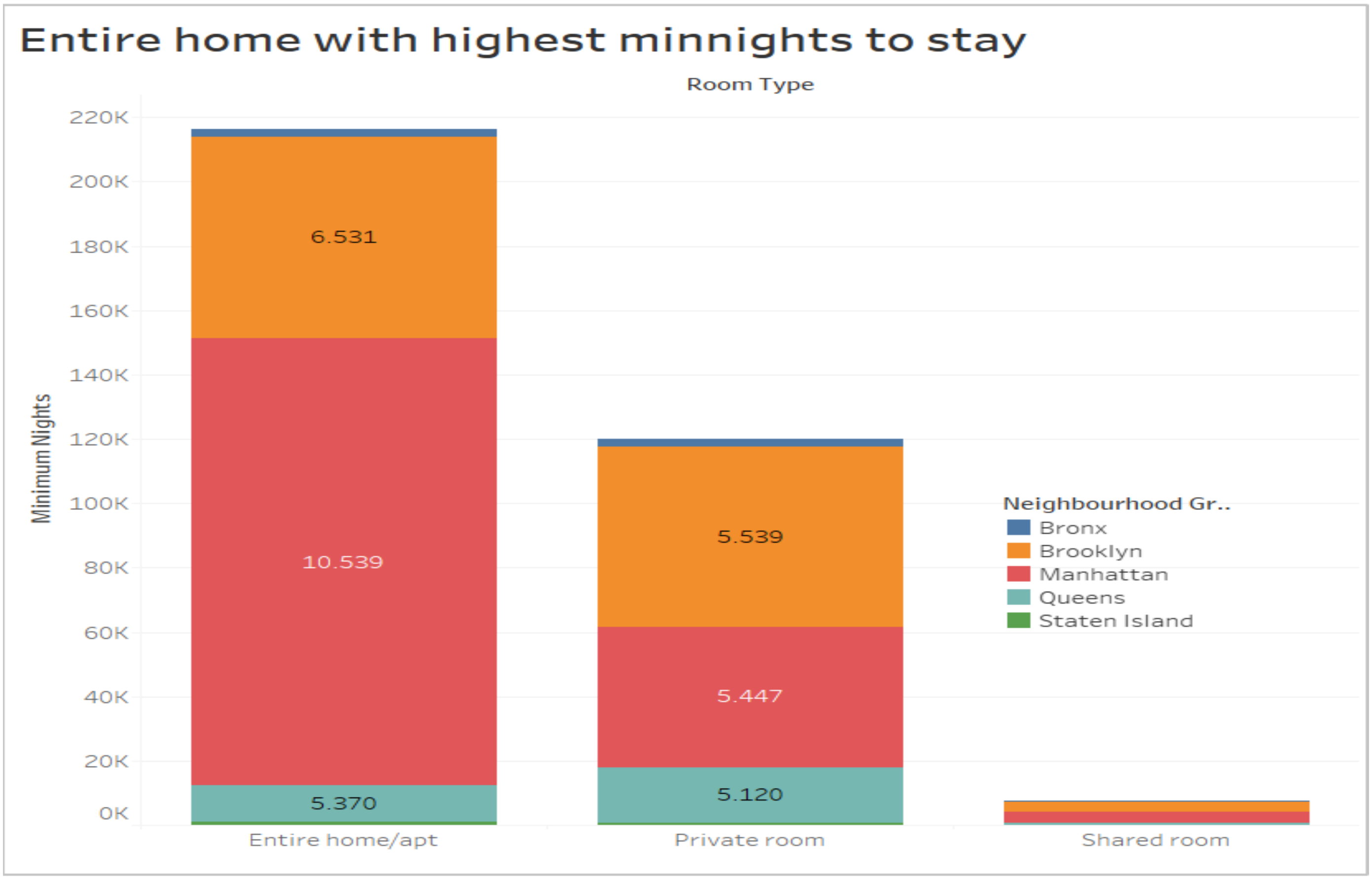
Graph:

Observation:

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.

4. Room types with highest minimum nights to stay:

Graph:



Observations:

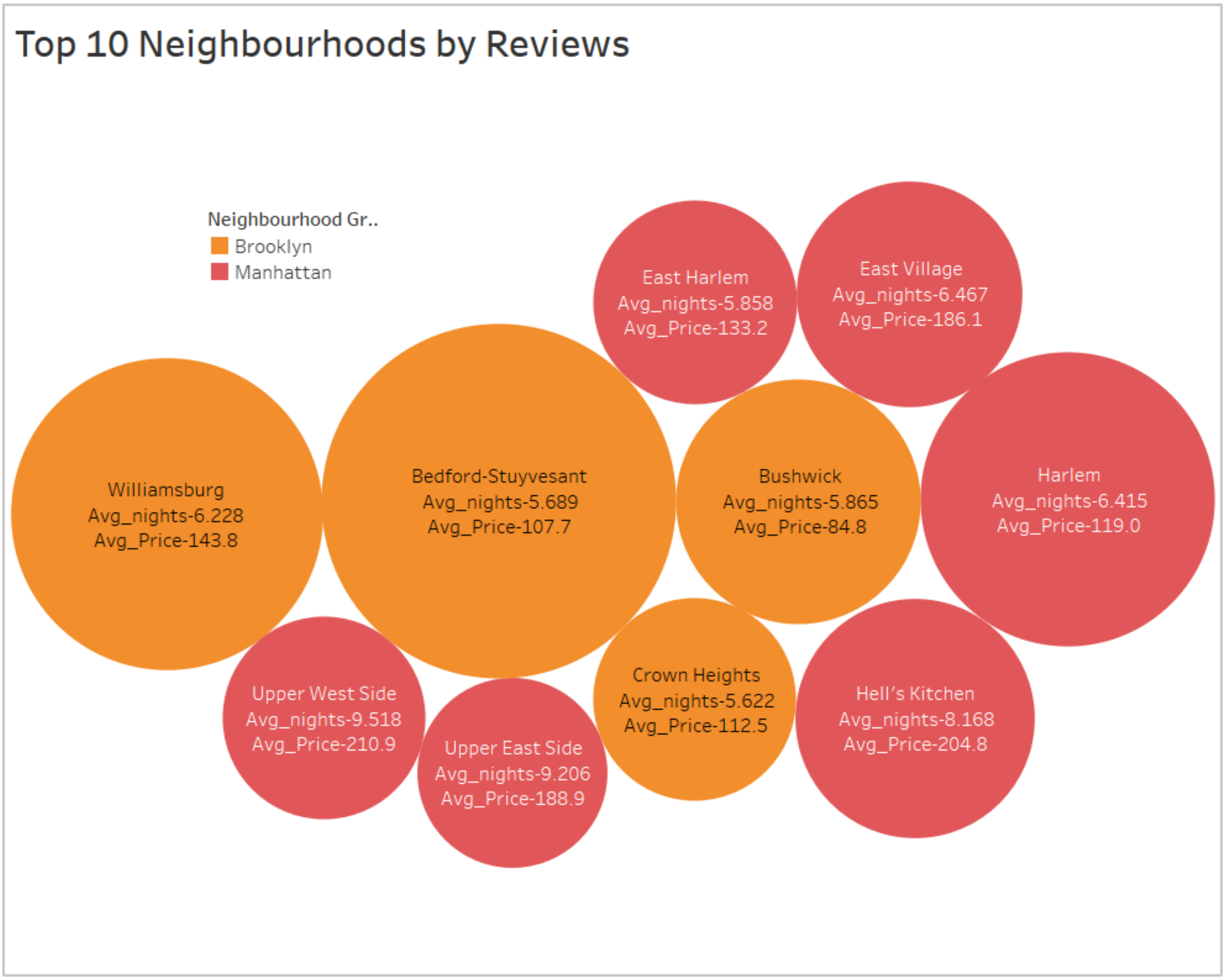
- **Entire home/apt types** are preferred more by the customers followed by Private rooms and then Shared Rooms. Mostly because they are also available for a higher number of minimum night's stay window booking as compared to Private and Shared rooms.
- **Manhattan** consist of **maximum Entire homes** and **private rooms** with **highest min nights** to stay.
- **Brooklyn** has **second most Entire homes** and **private rooms** with largest min nights to stay.

5. Top 10 Neighbourhoods by Reviews:

Observations:

- **Bedford-Stuyvesant, Williamsburg, Bushwick, Harlem and Crown heights, Hell's Kitchen, Upper west Side, Upper East Side, East Harlem, East Village are 10 topmost neighborhoods** by **reviews.**
- **Top 10** neighborhoods **belong** to **Manhattan** and **Brooklyn.**
- **Average price** offered ranges **110 -210 $** with **5-10 days** of **average min night's stay**

Graph:



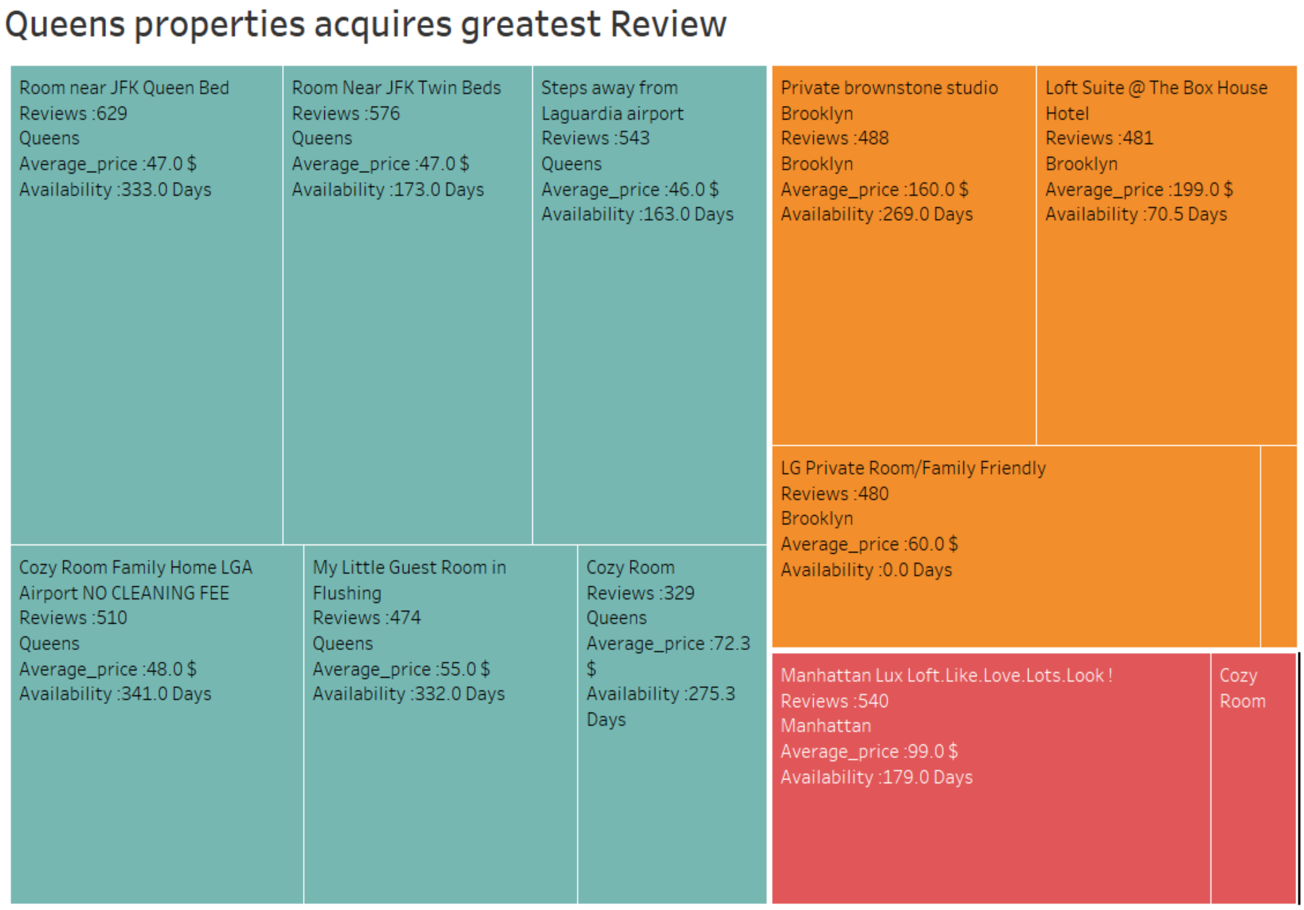
6. Top 10 Properties by Reviews:

Observations:

- **Rooms near JFK Queen Bed, Rooms near JFK Twin Beds, Cozy Room family home LGA, Steps away from LaGuardia airport, My little Guest Room, Cozy Room, Private brownstone studio, Loft Suite@The Box house hotel, LG private room, Manhattan Lux Loft Like.Love. Lots.Look!** are top 5 properties by reviews.

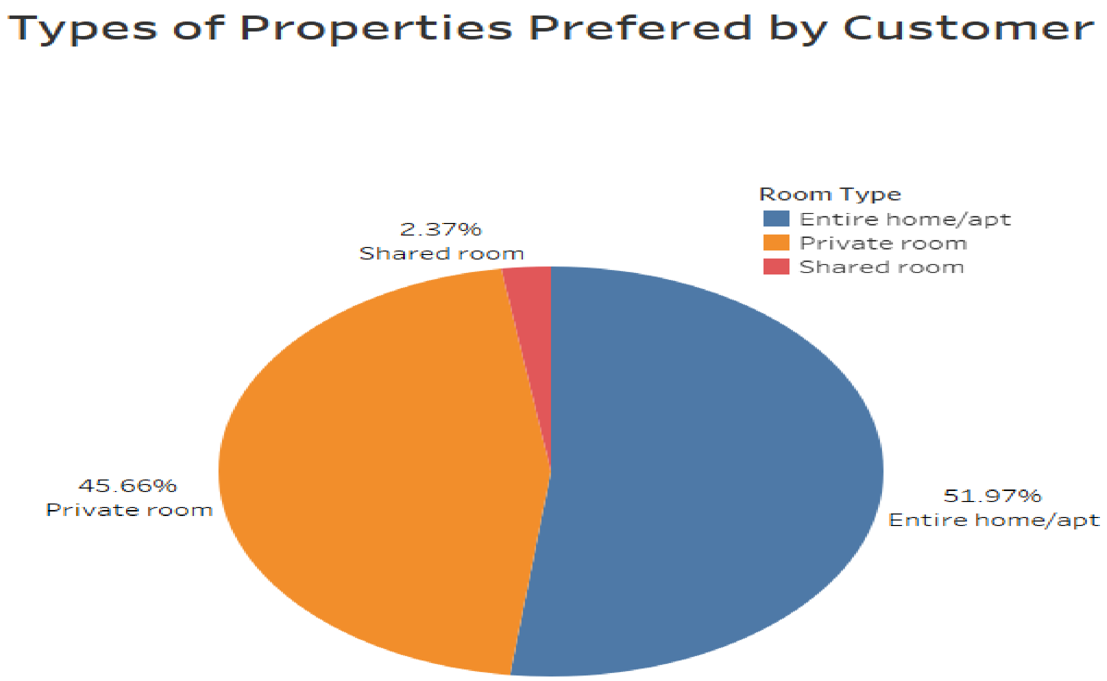
- Top 10 properties belong to **Queens** and **Brooklyn**.
- **Low Average** price around **40-60 $** offered by **Queens** properties with **availability 300+ days**.

Graph:



7. Types of Properties Preferred by Customer based on listings:

Graph:

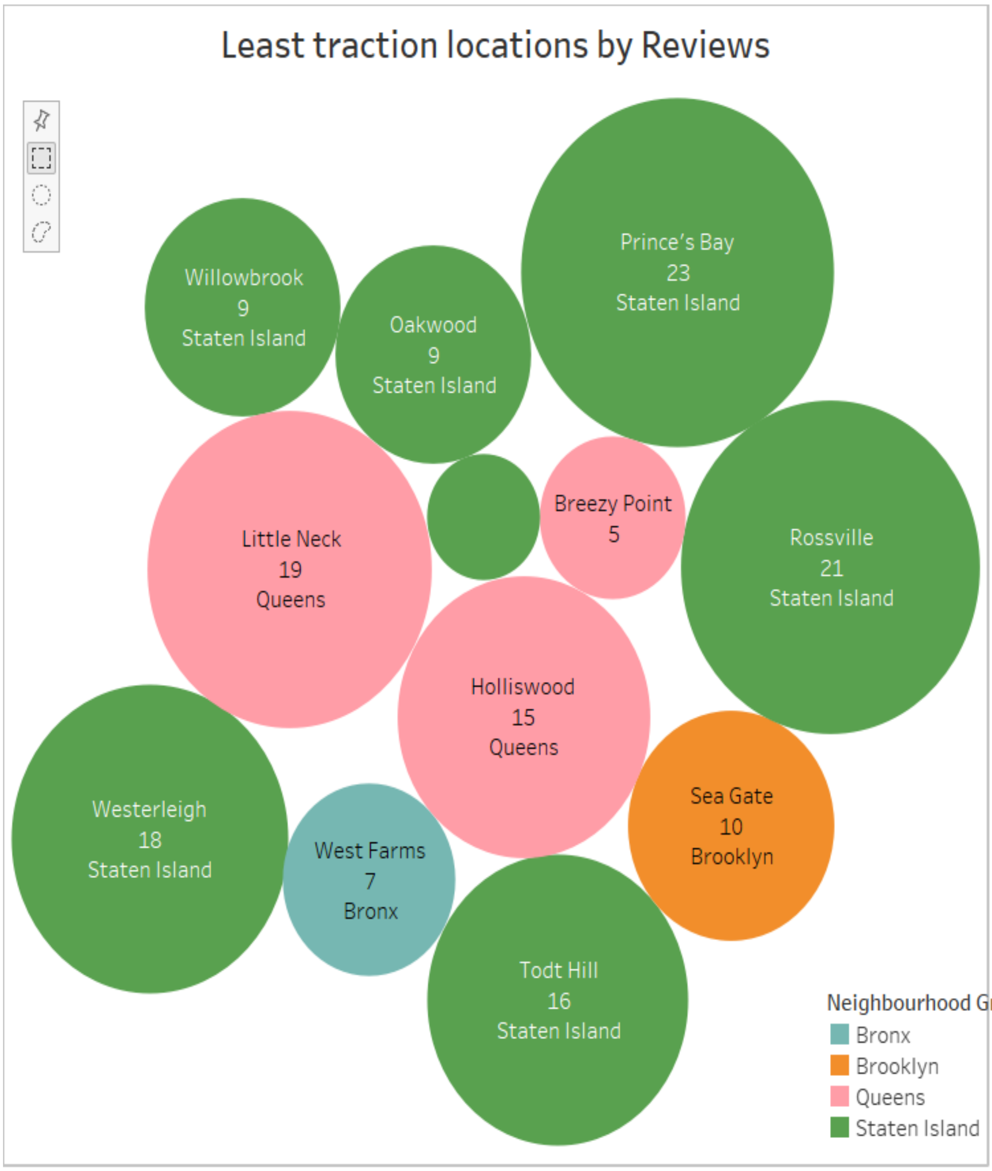


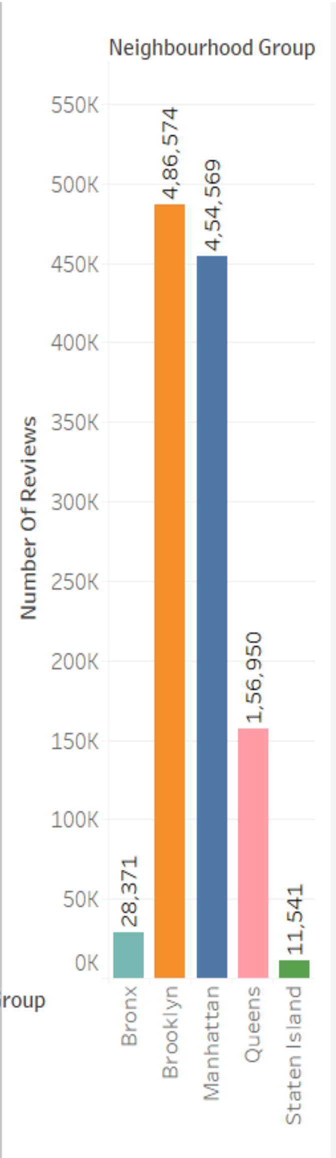
Observation:

- Customer prefers **Entire home** or **private rooms** most
- **Entire home/apt** contributes **51.9%** followed by **Private room** with **45.66%**
- Shared rooms account only 2.37%

8. Least Attracted Properties by reviews:

Graph:





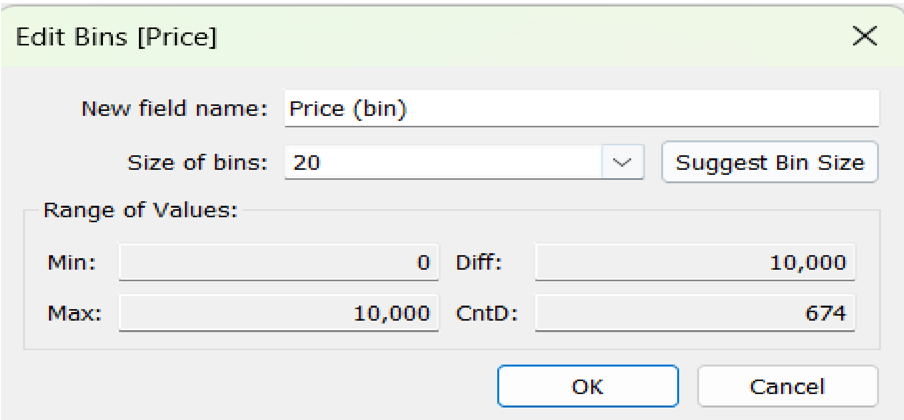
Observations:

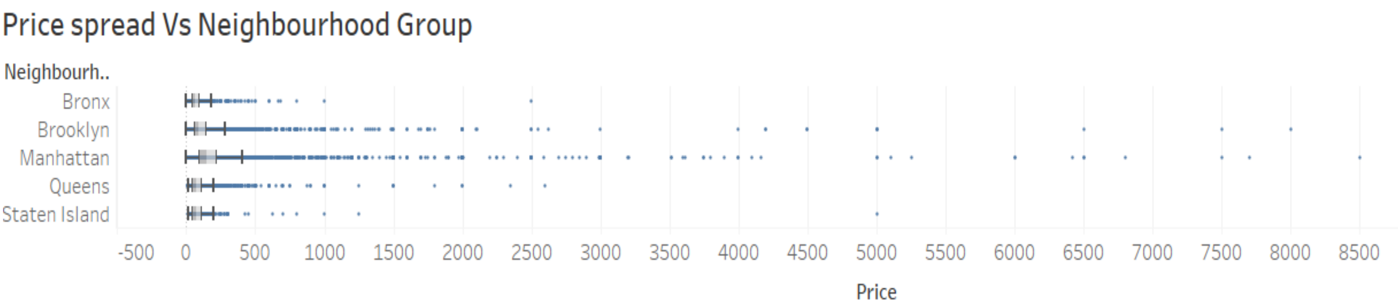
- *Staten Island* properties receive less reviews *11,544* from customers. Followed by Bronxs and *Queens* with reviews *28,871* and *1,56,950* respectively than others.
- Properties from *Bay terrace, Oakwood, Willowbrook* in *Staten Island* should make more customer oriented.
- *West Farms* from Bronxs and *Little Neck, Breezy point* and *Holiswood* from Queens properties should be followed next.

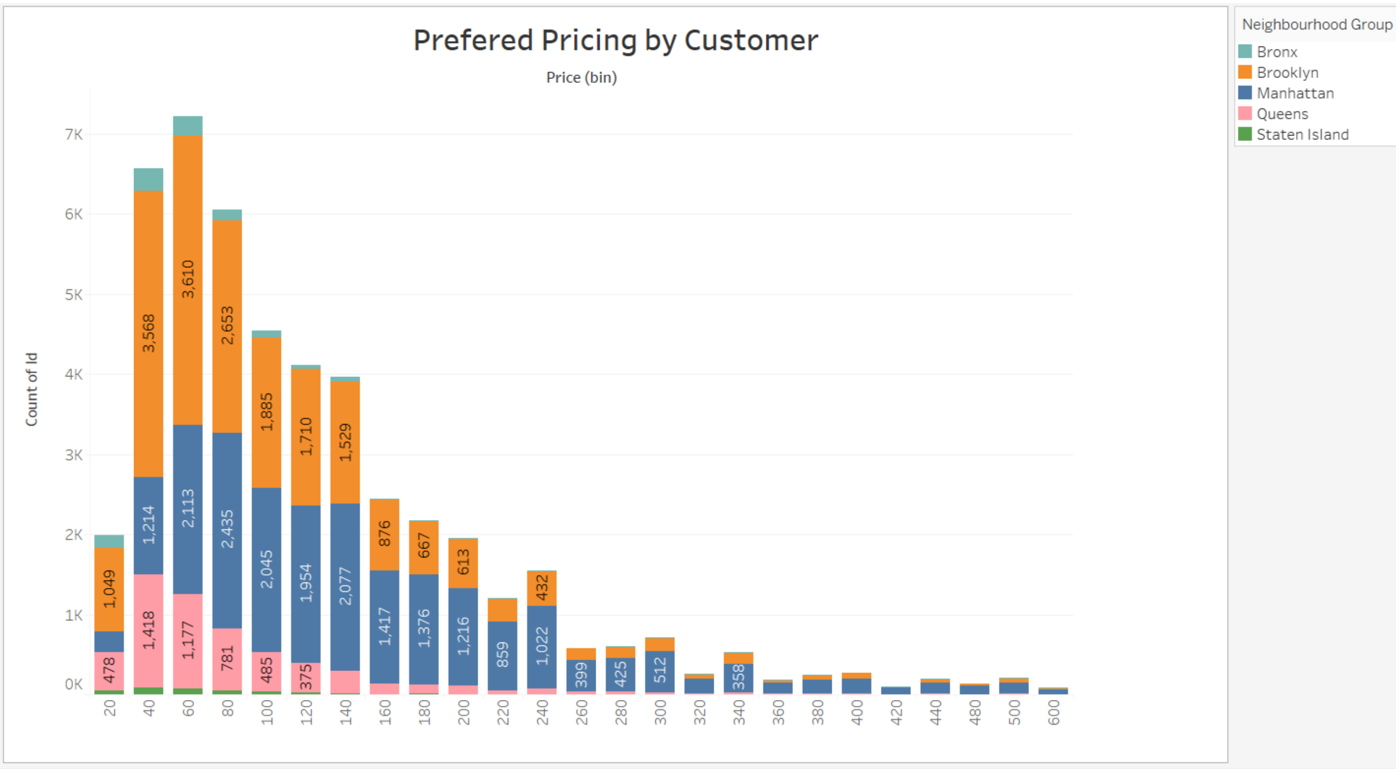
9. Preferred pricing with price spread:

Graphs:

We have created price bin of 20 here:







Observations:

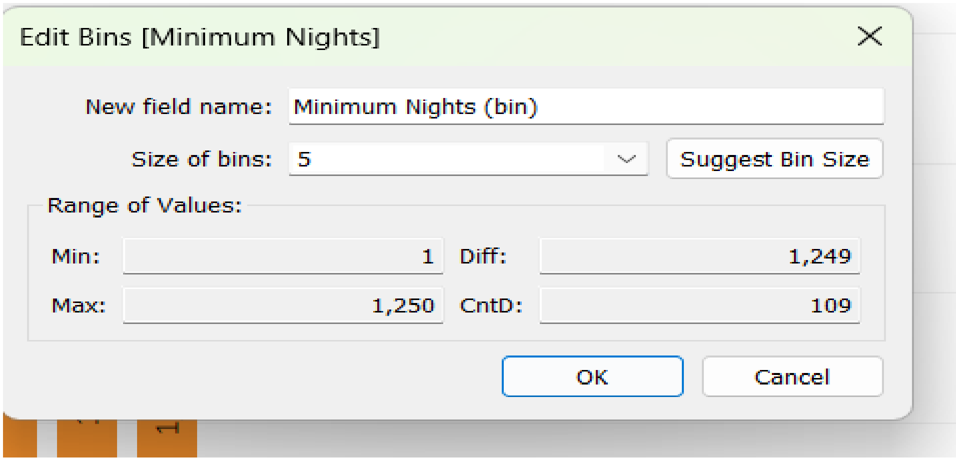
- Premium properties in Bronxs should be targeted as cost is already low. Non-Premium properties in Manhattan should be targeted as rates are high.
- Can be switch to 60-200 Pricing bucket as they are mostly preferred by customers.

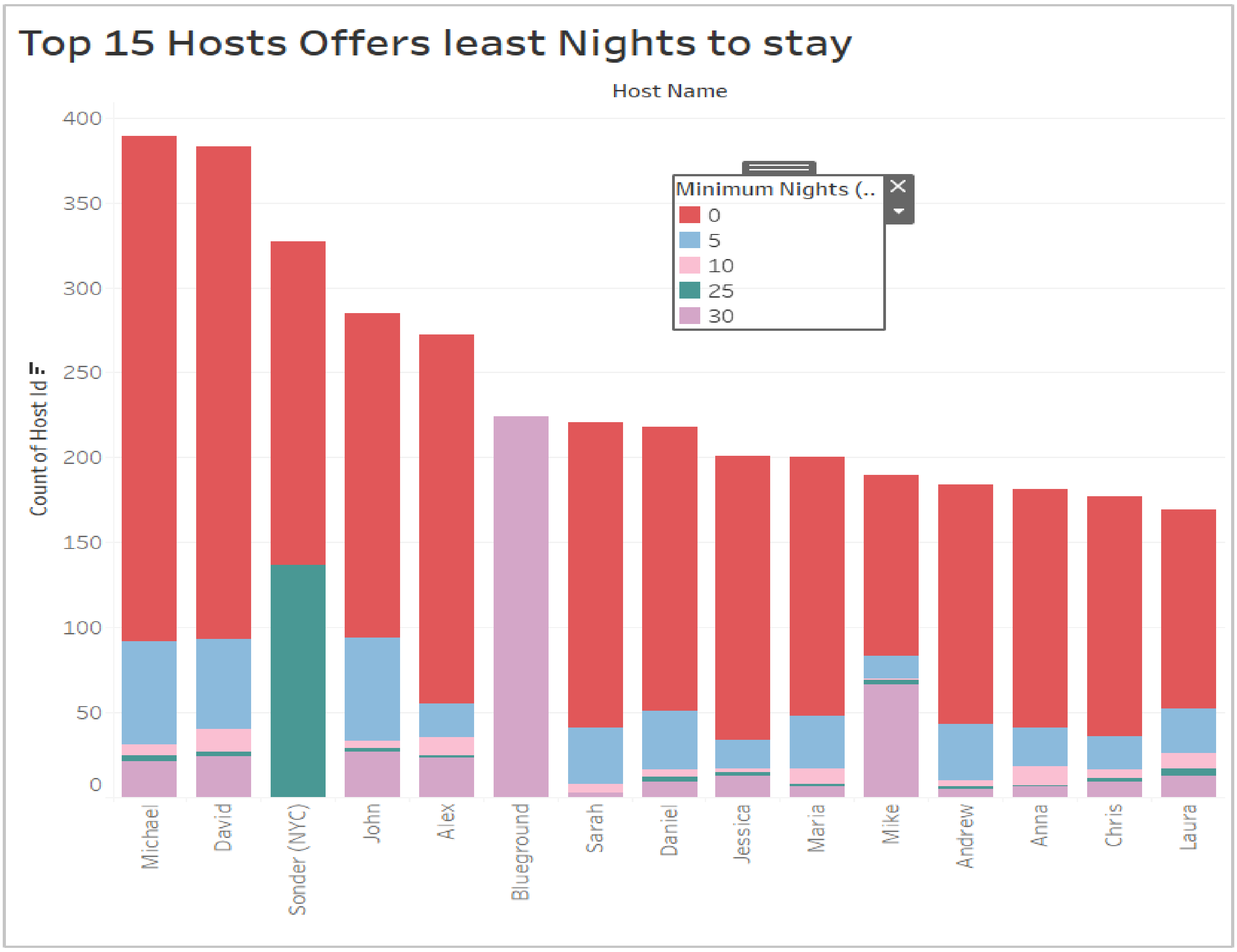
- Properties in Manhattan are most expensive with maximum pricing offered while Bronxs are least expensive.

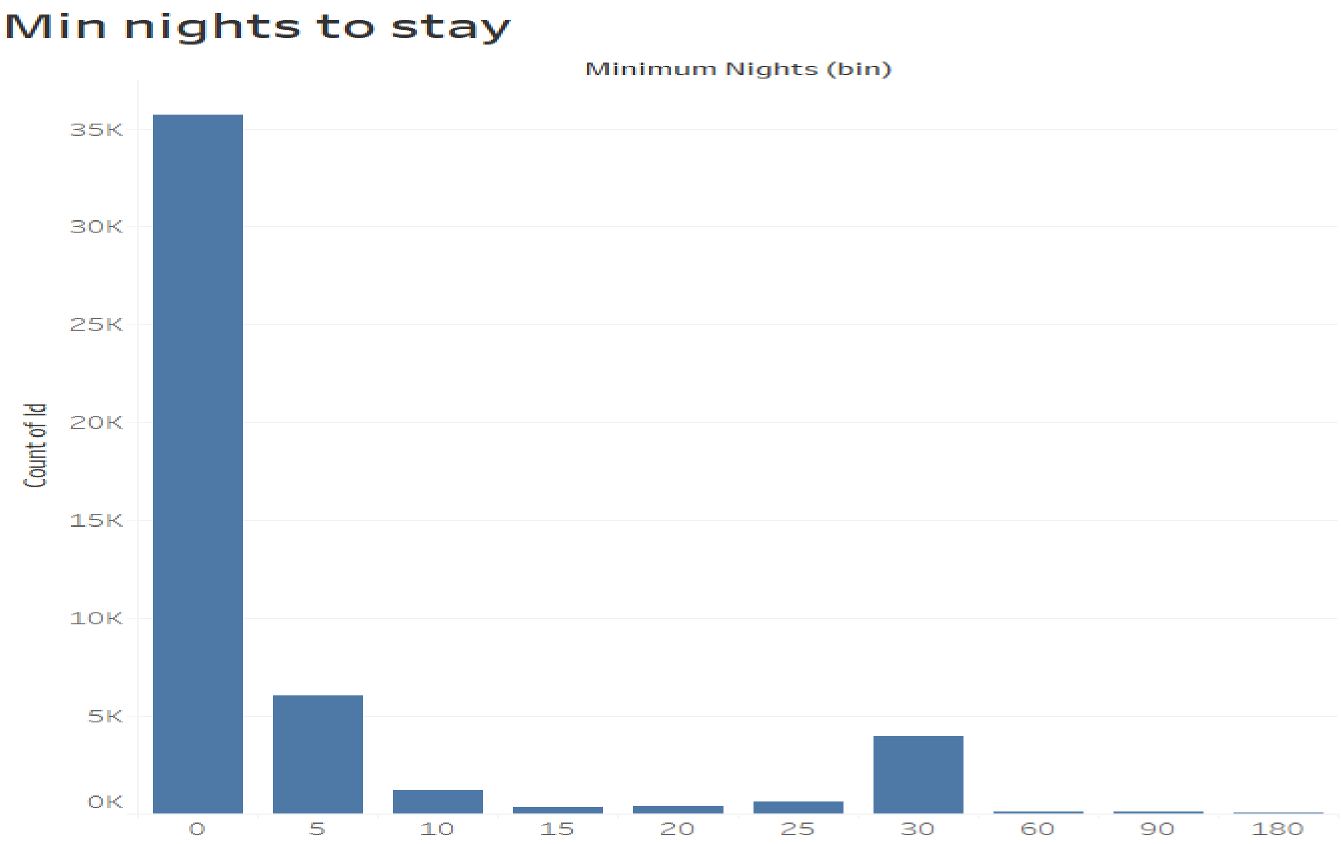
10.Top 15 Hosts with min nights to stay bucket:

Graphs:

We have created minimum nights bucket with size 5







Observations:

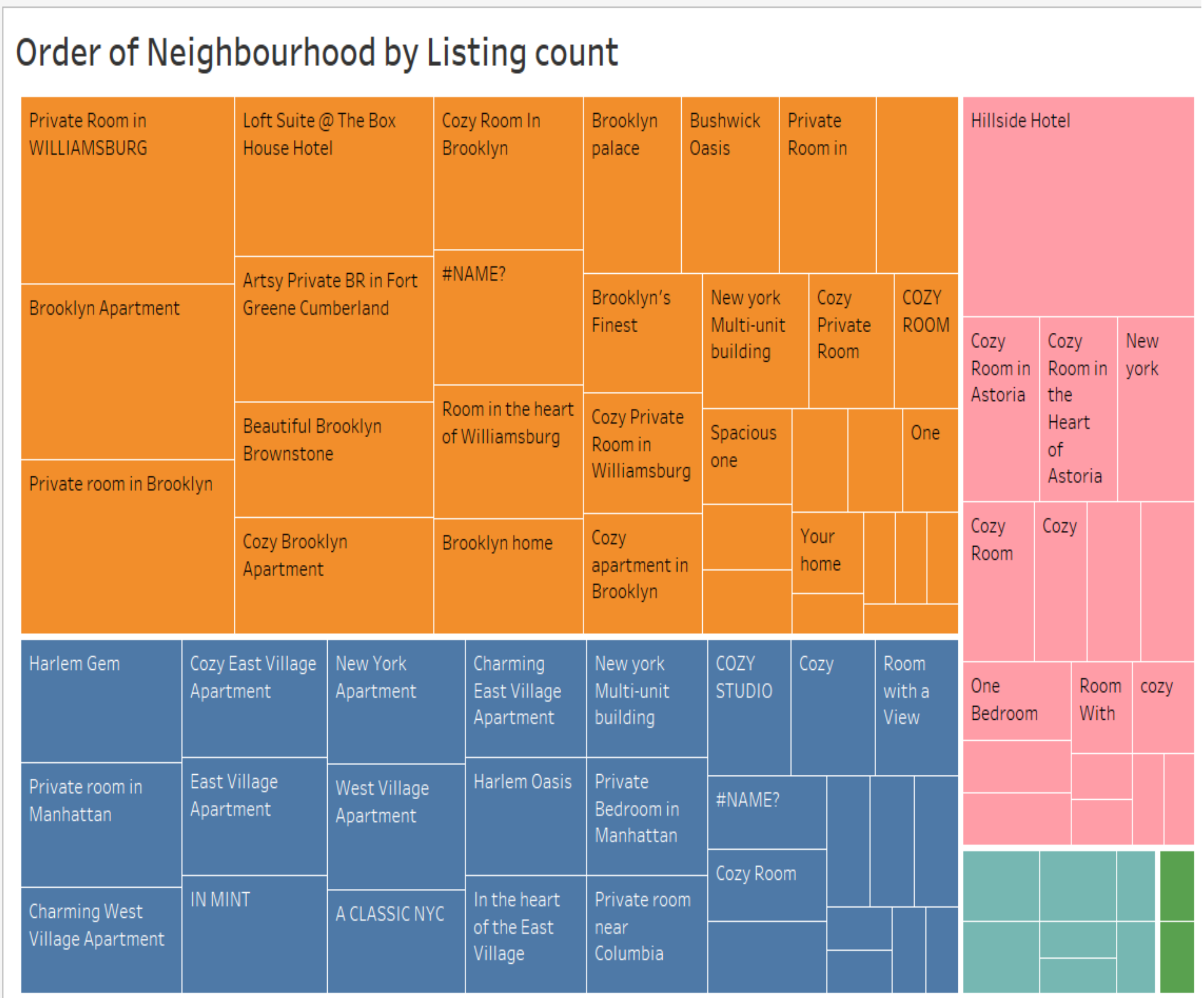
- For minimum nights to stay from **0-5 nights**, has maximum listing beyond **35K** in past.
- **Michael** is topmost listed host followed by **David** with we can have **one on one conversation to grow.**
- **Sonder (NYC)** and **Blueground** hosts are also offers **20-25** and **25-30 days** stay.

10.Neighbourhood vs Listing count

Observations:

- **Hillside hotel** listed most of times from **Queens** neighbourhood group.
- Private rooms in **Williamsburg** have maximum listing from **Brooklyn**.
- Harlem Gem followed by Cozy east village apt has maximum bookings from Manhattan

Graph:



4.Preparation of PPT 1:

- We have used graphs from bivariate analysis for PPT 1.
- We used graph 1 – 6 for this.
- Given detailed analysis of data with key findings.

5. Preparation of PPT 2:

- We have used graphs from bivariate analysis for PPT 1.
- We used graph 7-11 for this.
- Given decision-oriented insights with supported graphs.

6. Tools Used:

- Data Preparation and cleaning: **Jupyter Notebook**
- Data Visualization**: Tableau, Jupyter Notebook**
- Data Storytelling: **Power Point Presentation**