👤Team: K Jagannath, Amit Yadav & Pallavi Zadokar
🗓️ Date: 13/06/2023

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 that the restrictions have started lifting and people have started to travel more. Hence, Airbnb wants to make sure that it is fully prepared for this change.

2. Business Understanding

Airbnb is an American company based in San Francisco, California. It operates an online marketplace for lodging, primarily homestays for vacation rentals, and tourism activities. The platform is accessible via a website and mobile app.

Afterall, being an online marketplace for hosting personal homestays and private apartments in majority, the company had two types of customers. One who hosts their place and the another who books the place for a particular time is the end consumer utilizing the hosted place. Airbnb earns commission from both ends and hence have to make sure both of its customers are able to generate value from their business. They also have to make the hosted place offered on their platform provide the best services at reasonable prices and look out for the best technology to ease out the booking process for the end consumer without hassle.

3. Type of Data required to Analyze

Decline in the revenue could be for two major reasons, either the sites hosted on the platform are not able to provide better user experience or there could be a competitor in the market capturing the market share. Keeping the above in mind, we first try to work on the first reason as that is something internal to the company and can have the data in hand to identify the reasons behind the plummeting of the revenue.

Hence, we use the information of the hosted places on the platform to see where and what can be done to improve the end consumer experience. The data would majorly include the location and region of the hosted places, in our case we are targeting Borough (New York City) — the Bronx, Brooklyn, Manhattan, Queens and Staten Island, followed by their hosts details, prices of the hosted sites and reviews received by the end consumer.

4. Data Assumptions? (Assumption)

- Data Consistency: It is assumed that the dataset maintains consistency in the format and structure of the data across all records.
- Data Accuracy: It is assumed that the provided dataset is accurate and reliable, based on the assumption that Airbnb maintains accurate records of their listings in NYC.
- Data Privacy and Anonymity: It is assumed that appropriate measures have been taken to protect the privacy of hosts and guests in the dataset. Personal information has been removed to maintain data confidentiality.

5. Whom are we presenting?

- **Data Analysis Managers:** These people manage the data analysts directly for processes, and their technical expertise is basic.

- **Lead Data Analyst:** The lead data analyst looks after the entire team of data and business analysts and is technically sound.
- **Head of Acquisitions and Operations, NYC:** This head looks after all the property and hosts acquisitions and operations. Acquisition of the best properties, price negotiation, and negotiating the services the properties offer falls under the purview of this role.
- **Head of User Experience, NYC:** The head of user experience looks after the customer preferences and also handles the properties listed on the website and the Airbnb app. Basically, the head of user experience tries to optimize the order of property listing in certain neighbourhoods and cities in order to get every property the optimal amount of traction.

6. Recommendations

- Improve data collection: Gather additional details about amenities, features, and reviews for deeper insights.
- Segment customers: Identify different traveller categories and personalize marketing and recommendations.
- Leverage advanced analytics: Employ machine learning to predict preferences, demand, and optimize pricing.
- Acquire properties strategically in high-demand areas like Manhattan and Brooklyn, focusing on waterfront/coastal neighbourhoods.
- Diversify property offerings by acquiring more entire home / apartment listings and expanding the portfolio of private rooms.
- Negotiate favourable prices with hosts while considering the preferences of budget-conscious travellers.
- Enhance operational efficiency by allocating additional resources during peak months (June & July) and delivering exceptional guest experiences.
- Optimize property listing in order to highlight popular areas like Manhattan while considering affordability in neighbourhoods like the Bronx.
- Personalize property recommendations based on customer preferences, emphasizing features like waterfront locations and exclusive amenities.
- Improve visibility for less popular properties through curated lists, promotions, and enhanced descriptions to attract a wider range of customers.
- Segment customers based on their preferences, such as budget-conscious travellers, those seeking private stays, or those looking for shorter stays.
- Identify target neighbourhoods based on their popularity, affordability, and unique offerings to enhance the user experience.

B. Method of Analysis along with code:

1. Data Understanding and Preparation

Before we started the basic understanding of the data in hand, we imported relevant libraries available in Python. Below are the libraries that we imported,

```
# Importing Libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
plt.style.use('seaborn-dark-palette')
from scipy import stats
```

```
import datetime as dt
import plotly
import plotly.express as px

import warnings
warnings.filterwarnings("ignore")

pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
pd.set_option('display.float_format', lambda x: '%.3f' % x)
```

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 dataframe. Below is the code used for the same.

```
# read the dataset
airbnb = pd.read_csv("AB_NYC_2019.csv")
airbnb.head()


# Dimensions
airbnb.shape
airbnb.info()
airbnb.describe()
```

2. Features in the dataframe:

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,

3. Handling Missing Values and Outliers:

We dropped the ID column as it is not important for analysis.

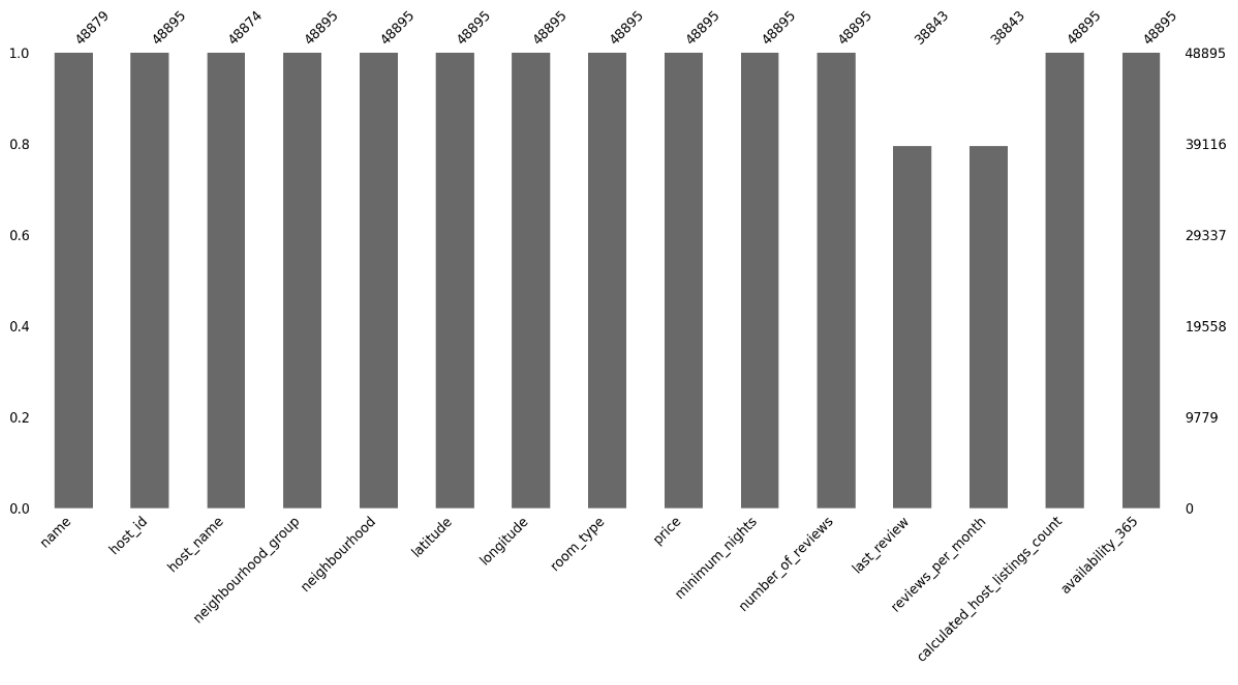
```
# dropping ‘id’ column
airbnb.drop('id',axis= 1 ,inplace=True)
```

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

- After analyzing, it seems like these values (host name and their place name) are missing by chance, hence we need to collect this information from the Host acquisition and operations team.
- Finally, we just imputed the missing values of reviews_per_month with a 0.

Below is the code we used to identify missing values. We also imported missingno library to do the same.

```
# user defined function to calculate missing values
def Missing_Percentage(dataFrame):
     Missing_perc
100*(dataFrame.isna().mean()).sort_values(ascending=False)
    return missing_perc

# Numerical interpretation using user defined function
Missing_Percentage(airbnb)

# missing values graphical representation for each column
import missingno as mno
print("Missing Number Graph for airbnb data")
mno.bar(airbnb)
plt.show()
# Full grey bar means no missing values,
# Higher grey bar means less missing value,
# Smaller grey bar (white part) means more missing values
```

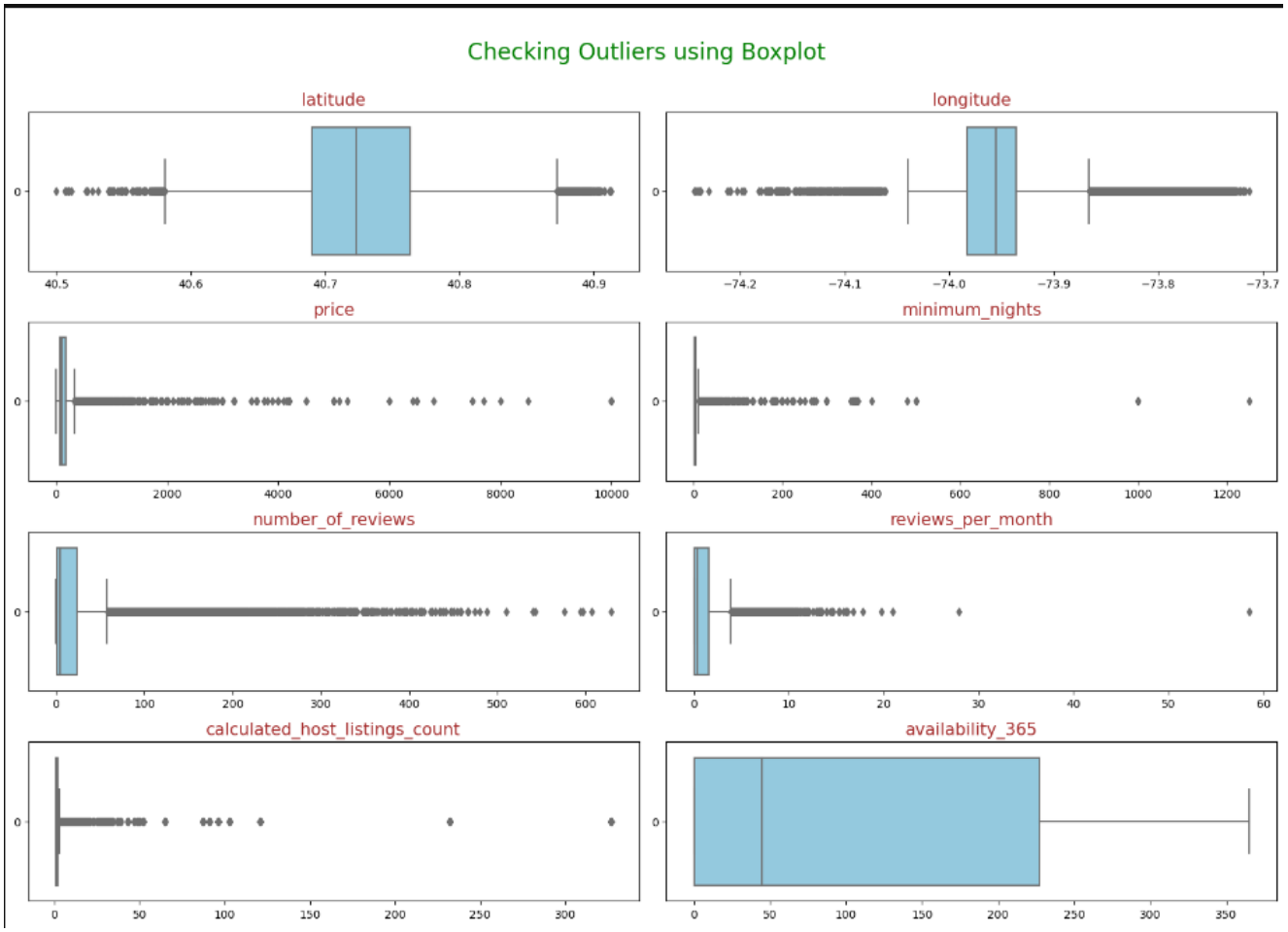


Post analyzing and treating the missing values accordingly we treated the spread in the data frame i.e. outliers. Below is the code we used to identify the spread of the outliers.

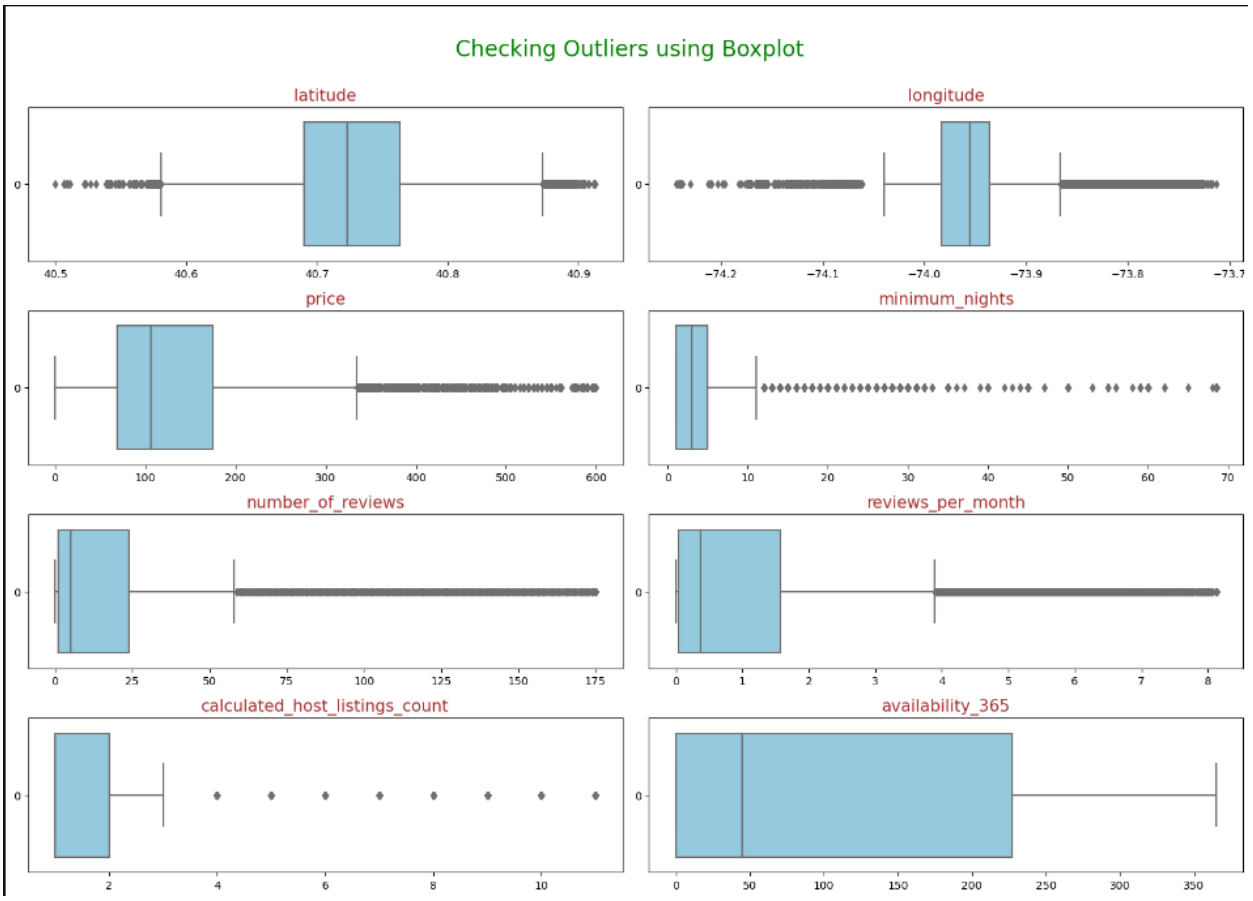
```
num_cols = airbnb.select_dtypes(include=['int64', 'float64']).columns
```

Handling Outliers

```
# UDF for checking Outliers
def Check_Outliers(data,columnList):

    plt.figure(figsize=([16,14]))
    plt.subplots_adjust(wspace=0.4,hspace=0.5)

    for i,j in enumerate(columnList):
        plt.subplot(5,2,i+1)
        sns.boxplot(data[j], orient = "h", color="skyblue")
        plt.xlabel(None)
        plt.ylabel(None)
                                    plt.suptitle("\nChecking    Outliers    using
Boxplot\n",fontsize=20,color="green",)
        plt.title(j,fontsize=15,color='brown')
        plt.tight_layout()
```



```
## Fixing (price , minimum_nights , number_of_reviews , reviews_per_month , calculated_host_listings_count )
# Taking 0.1 and 0.9 as lower and upper bound threshold for flooring & capping
# Defining UDF to treat outliers via capping and flooring

def Outlier_treatment(df,columnList):
    for i in columnList:
        q1 = df[i].describe(percentiles=[0.1,0.9])["10%"]
        q3 = df[i].describe(percentiles=[0.1,0.9])["90%"]
        IQR = q3 - q1

        upper_bound = q3 + 1.5*IQR
        lower_bound = q1 - 1.5*IQR

        # capping upper_bound
        df[i] = np.where(df[i] > upper_bound, upper_bound,df[i])

        # flooring lower_bound
        df[i] = np.where(df[i] < lower_bound, lower_bound,df[i])
capping_cols = ['price' , 'minimum_nights' , 'number_of_reviews' , 'reviews_per_month' , 'calculated_host_listings_count']

# UDF
Outlier_treatment(airbnb,capping_cols)
# Checking outliers using boxplot after fixing them

# UDF for boxplot
Check_Outliers(airbnb,num_cols)
```



4. Feature Scaling / 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 codes of all the variables that were engineered for our further analyses.

```
# Binning for 'minimum_nights' into a new column 'minimum_nights_grp'

bins=np.arange(0,71,10)
grp= ["<10", "10 to 20", "20 to 30", "30 to 40", "40 to 50", "50 to 60", "60+"]

airbnb["minimum_nights_grp"] = pd.cut(airbnb['minimum_nights'],bins,labels=grp)

# Also converting the 'minimum_nights_grp' to object dtype
airbnb['minimum_nights_grp'] = airbnb['minimum_nights_grp'].astype(str)
```

Similarly we did for rest of the numerical columns.

This is the dataset information after all data cleaning and feature engineering

```
1  airbnb.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48895 entries, 0 to 48894
Data columns (total 21 columns):
 #   Column                              Non-Null Count  Dtype
---  ------                              --------------  -----
 0   name                                48879 non-null  object
 1   host_id                             48895 non-null  object
 2   host_name                           48874 non-null  object
 3   neighbourhood_group                 48895 non-null  object
 4   neighbourhood                       48895 non-null  object
 5   latitude                            48895 non-null  float64
 6   longitude                           48895 non-null  float64
 7   room_type                           48895 non-null  object
 8   price                               48895 non-null  float64
 9   minimum_nights                      48895 non-null  float64
 10  number_of_reviews                   48895 non-null  float64
 11  last_review                         48895 non-null  object
 12  reviews_per_month                   48895 non-null  float64
 13  calculated_host_listings_count      48895 non-null  float64
 14  availability_365                    48895 non-null  int64
 15  minimum_nights_grp                  48895 non-null  object
 16  number_of_reviews_grp               48895 non-null  object
 17  reviews_per_month_grp               48895 non-null  object
 18  calculated_host_listings_count_grp  48895 non-null  object
 19  availability_365_grp                48895 non-null  object
 20  price_grp                           48895 non-null  object
dtypes: float64(7), int64(1), object(13)
memory usage: 7.8+ MB
```

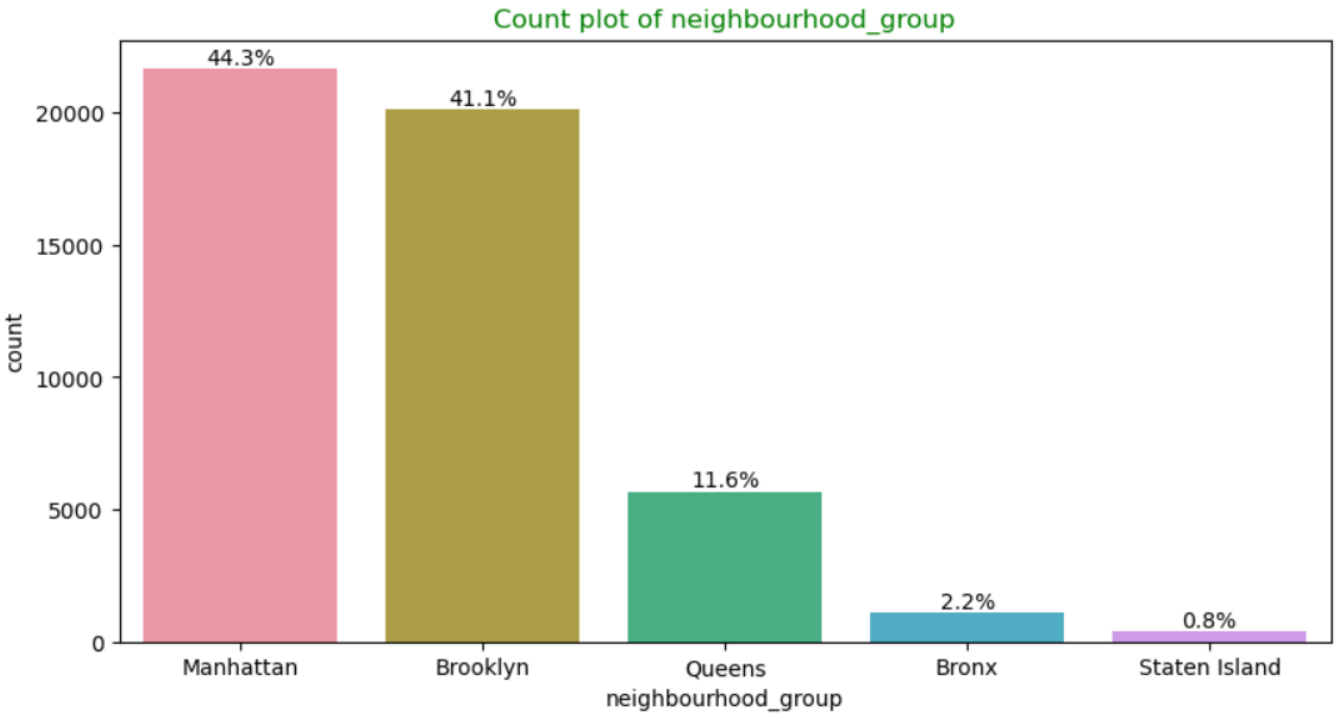
We also made the ***Revenue = price * minimum_nights*** column in Tableau calculated field

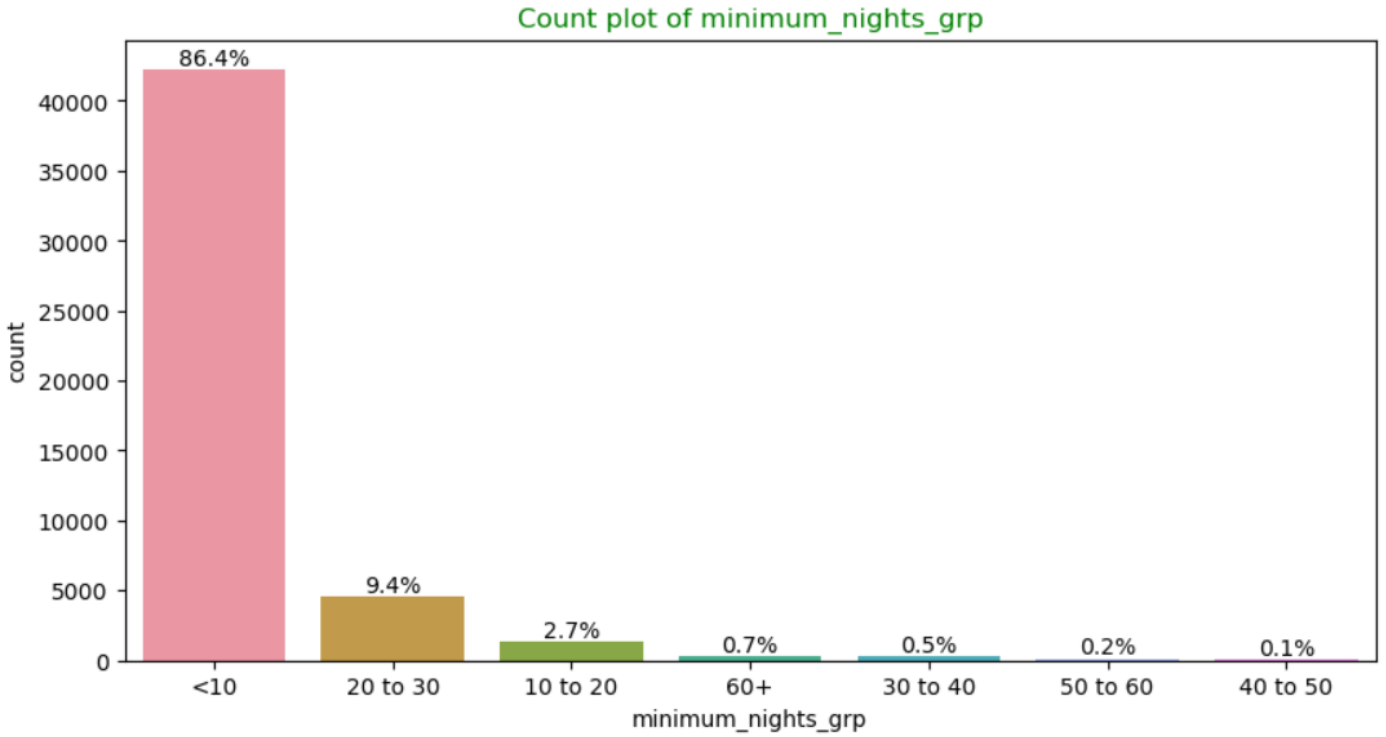
5. Exploratory Data Analysis

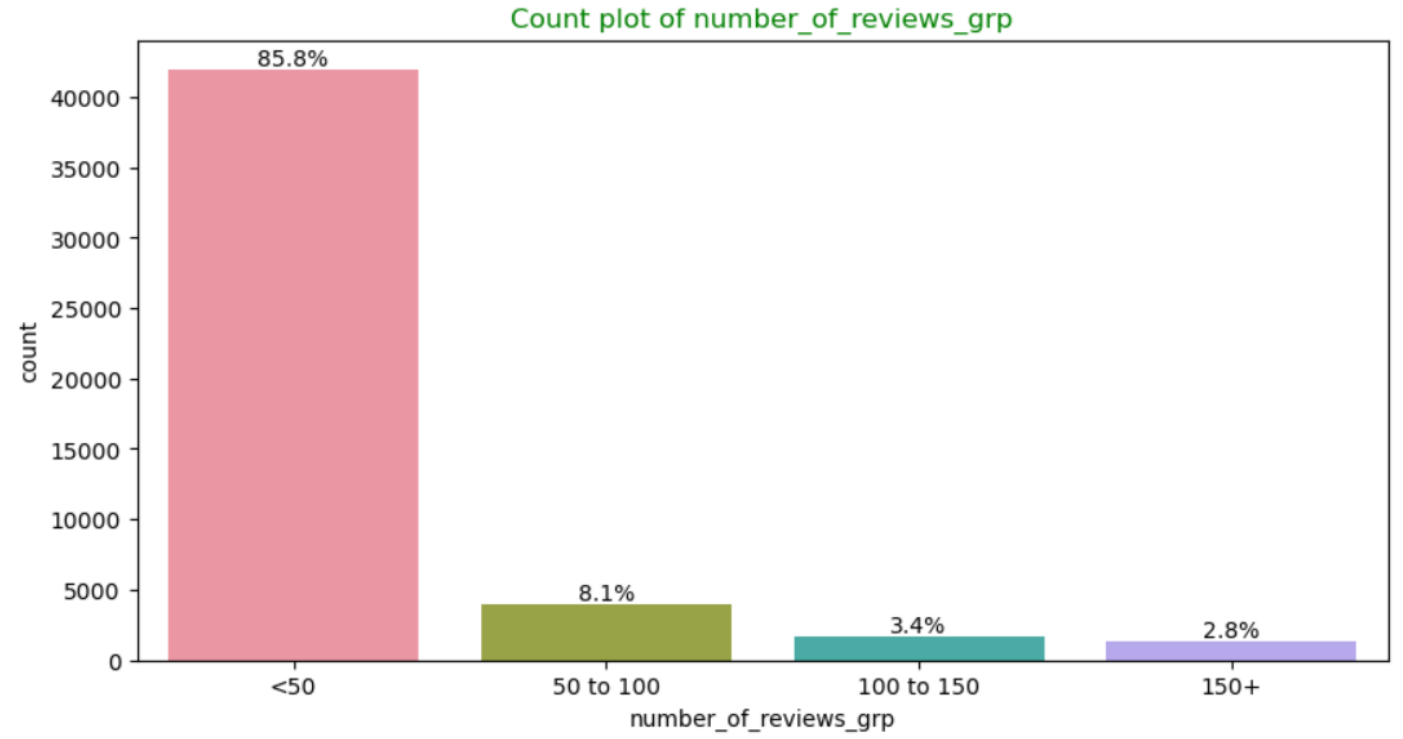
- Univariate Analysis of Categorical Columns

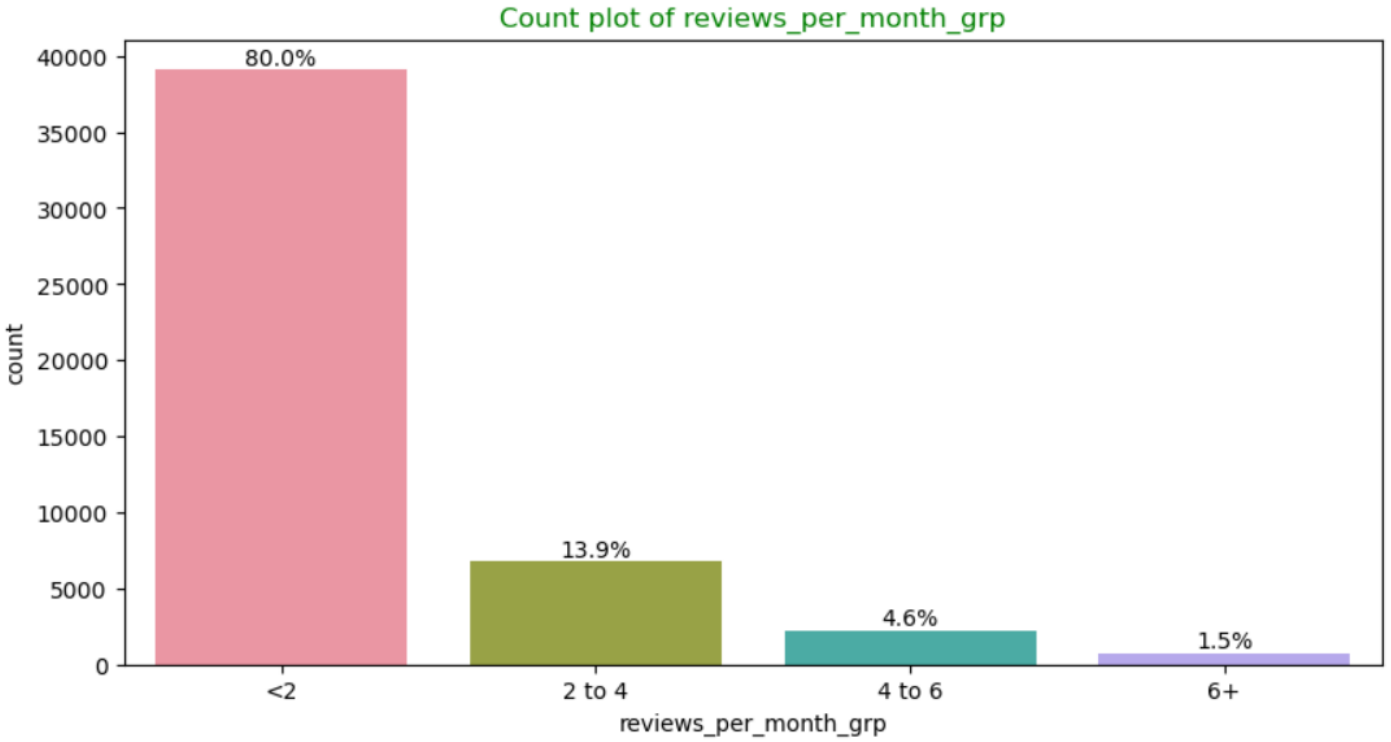
```
cat_cols = ['neighbourhood_group','room_type','minimum_nights_grp',
            'number_of_reviews_grp','reviews_per_month_grp',
            'availability_365_grp','price_grp',
            'calculated_host_listings_count_grp']

num_cols = ['price','minimum_nights','number_of_reviews','availability_365']
```

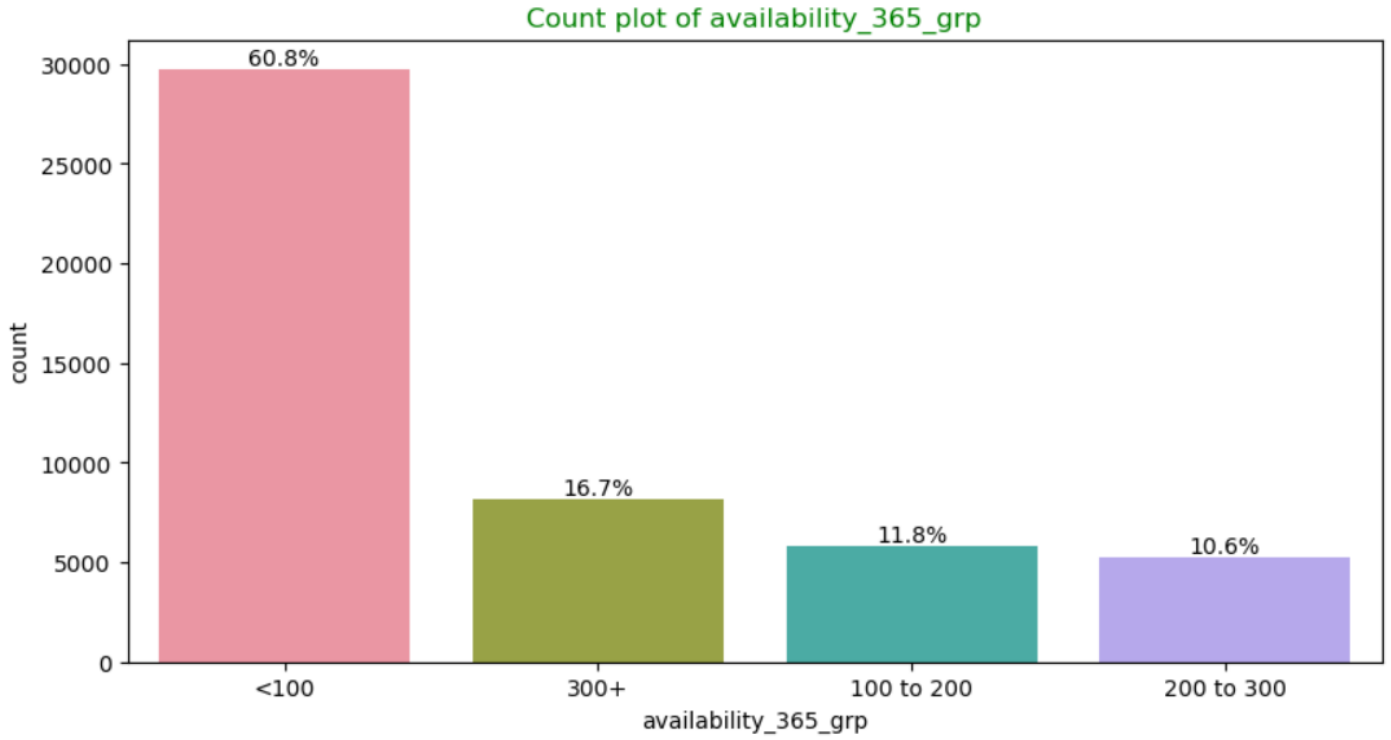
```
# Define the colors for the blue-teal color scheme
color_left = "#004E64"  # Left side color (e.g., darker teal)
color_right = "#00B1B5"  # Right side color (e.g., lighter teal)

# Create the blue-teal color palette
blue_teal_palette = sns.color_palette([color_left, color_right])

# Set the color palette
sns.set_palette(blue_teal_palette)

# Countplot of columns with value_counts percentage as annotation
for i in cat_cols:
    plt.figure(figsize=[10, 5])
    plt.title("Count plot of {}".format(i), color="green")
    large_to_small = airbnb.groupby(i).size().sort_values().index[::-1]
    ax = sns.countplot(x=i, data=airbnb, order=large_to_small)
    total = len(airbnb[i])
    plt.xticks(rotation=0)

    for p in ax.patches:
        text = '{:.1f}%'.format(100 * p.get_height() / total)
        x = p.get_x() + p.get_width() / 2.
        y = p.get_height()

          ax.annotate(text, (x, y), ha='center', va='center', xytext=(0, 5),
textcoords='offset points')
```

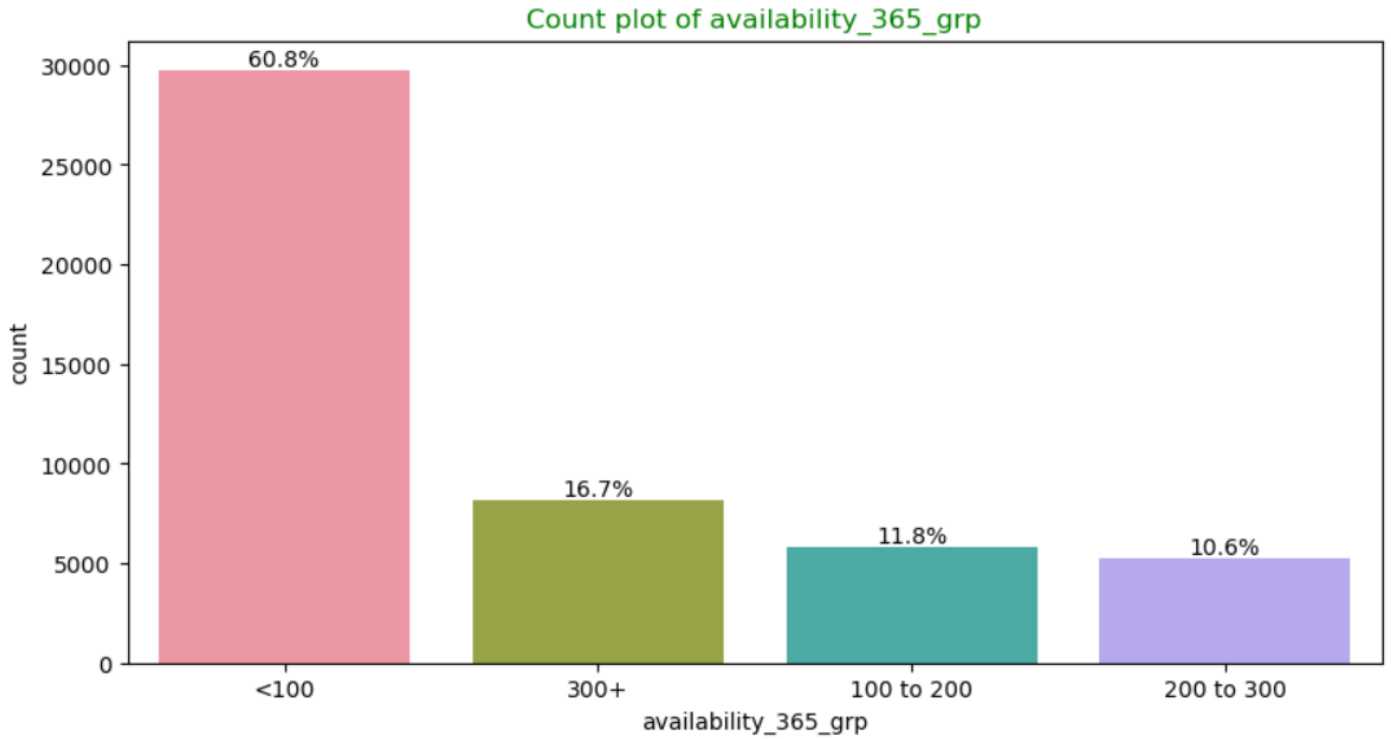


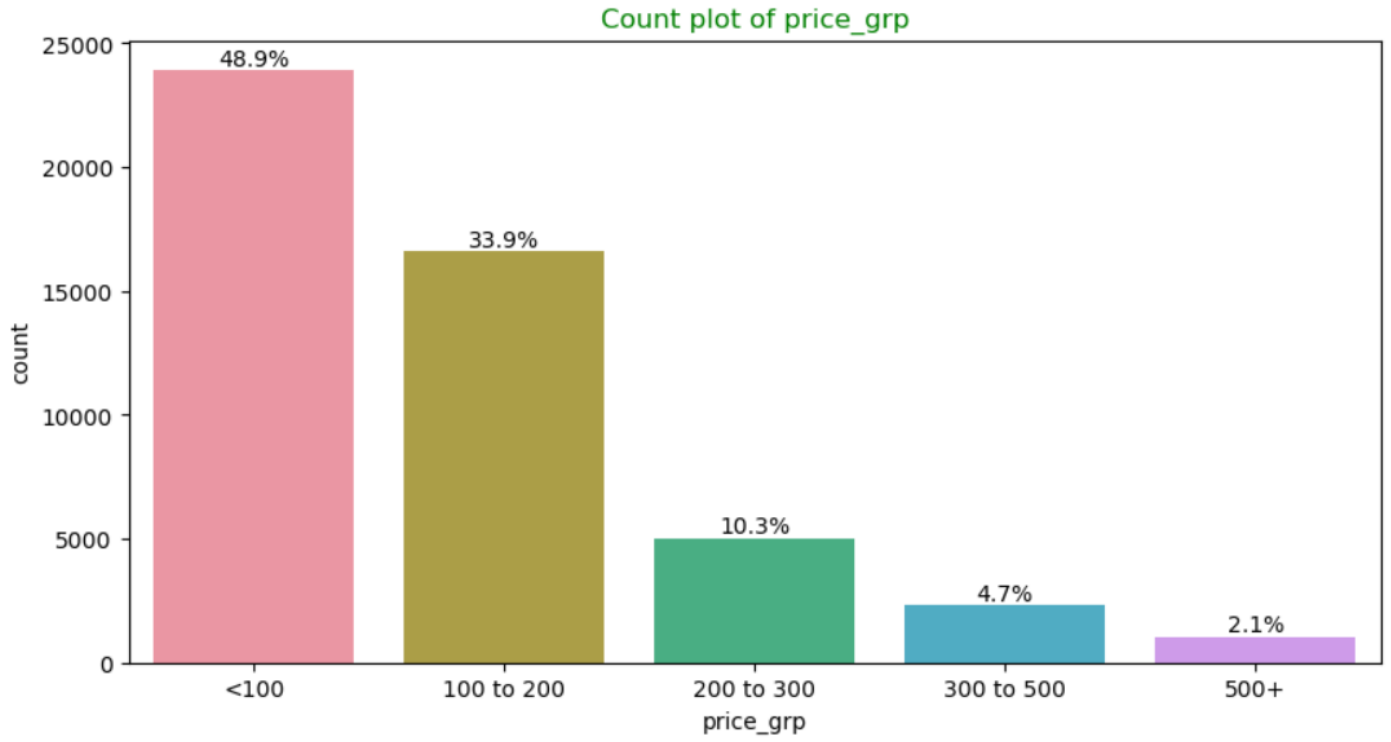

Count plot of minimum_nights_grp
86.4%
9.4%
2.7%
0.7%
0.5%
0.2%
0.1%
count
0
5000
10000
15000
20000
25000
30000
35000
40000
<10
20 to 30
10 to 20
60+
30 to 40
50 to 60
40 to 50
minimum_nights_grp


Count plot of number_of_reviews_grp
85.8%
8.1%
3.4%
2.8%
count
0
5000
10000
15000
20000
25000
30000
35000
40000
<50
50 to 100
100 to 150
150+
number_of_reviews_grp


Count plot of reviews_per_month_grp
count
reviews_per_month_grp
80.0%
13.9%
4.6%
1.5%
<2
2 to 4
4 to 6
6+
0
5000
10000
15000
20000
25000
30000
35000
40000

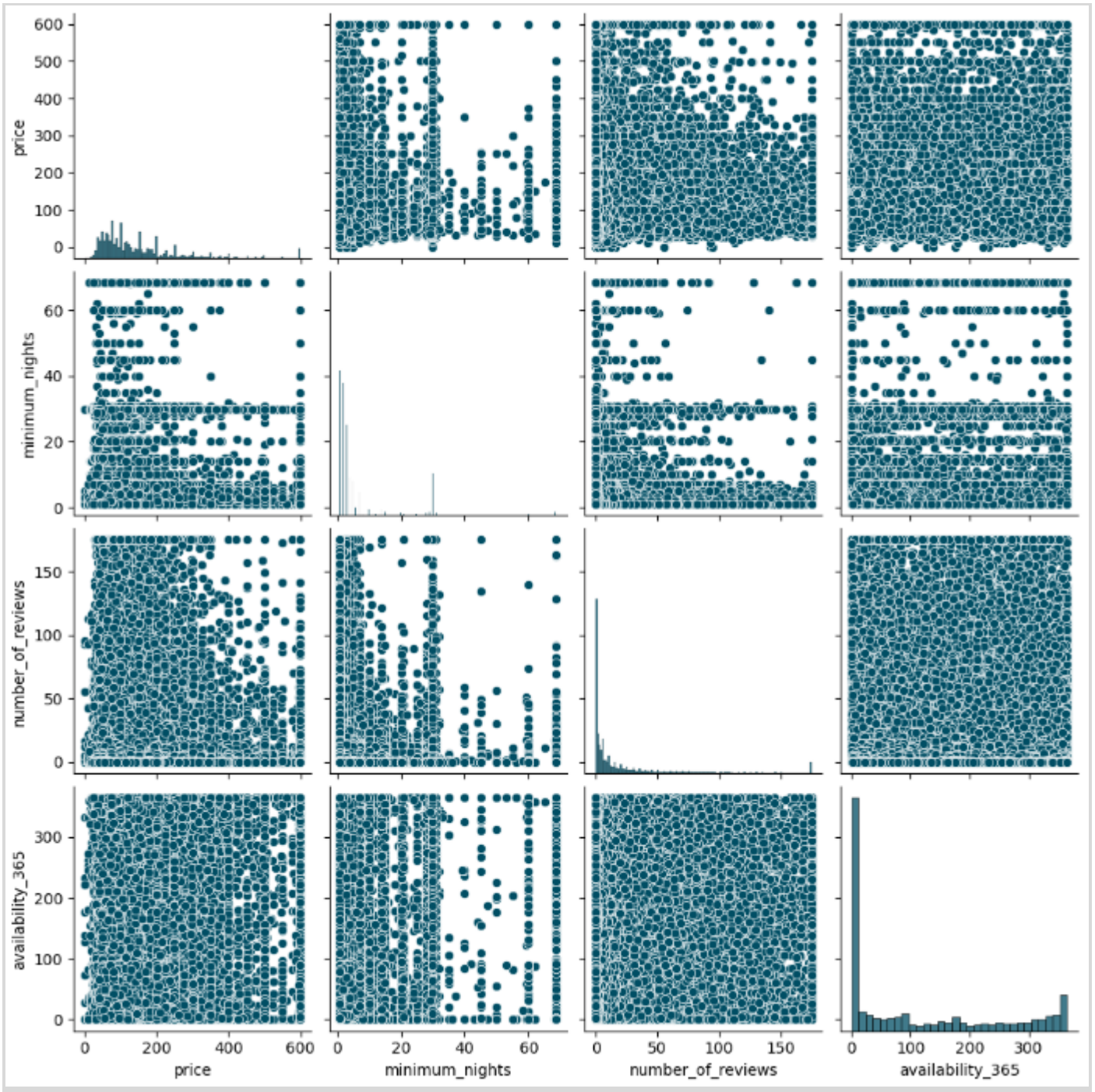

Count plot of availability_365_grp
count
availability_365_grp
60.8%
16.7%
11.8%
10.6%
<100
300+
100 to 200
200 to 300
0
5000
10000
15000
20000
25000
30000


Count plot of availability_365_grp
60.8%
16.7%
11.8%
10.6%
count
availability_365_grp
<100
300+
100 to 200
200 to 300

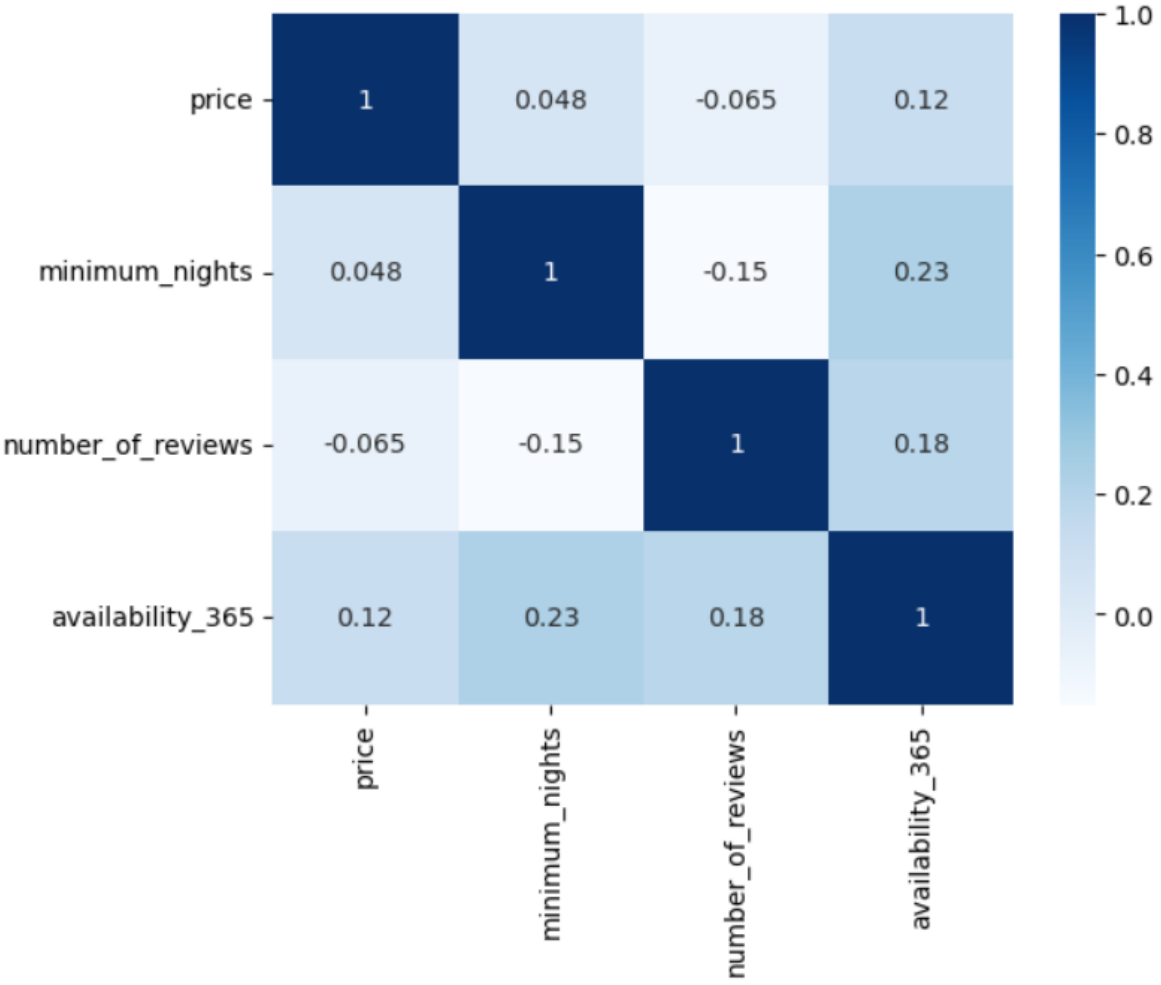

Count plot of price_grp
48.9%
33.9%
10.3%
4.7%
2.1%
count
price_grp
<100
100 to 200
200 to 300
300 to 500
500+

- Bivariate Analysis for Numerical Variables

```
plt.figure(figsize=(16, 4))
sns.pairplot(data=airbnb,vars=num_cols)
plt.show()
```

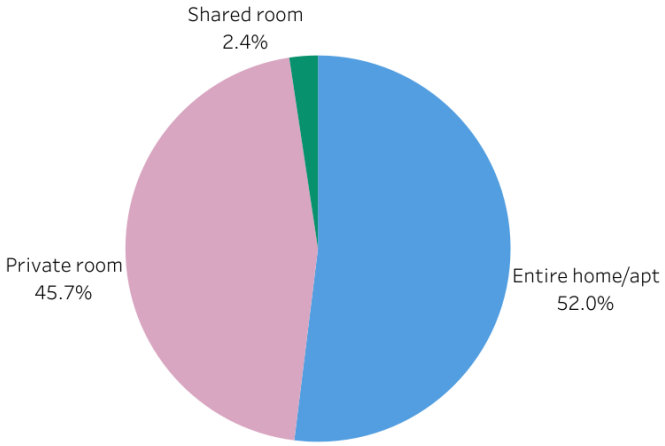


- Correlation Plot



No correlation found between the above features.

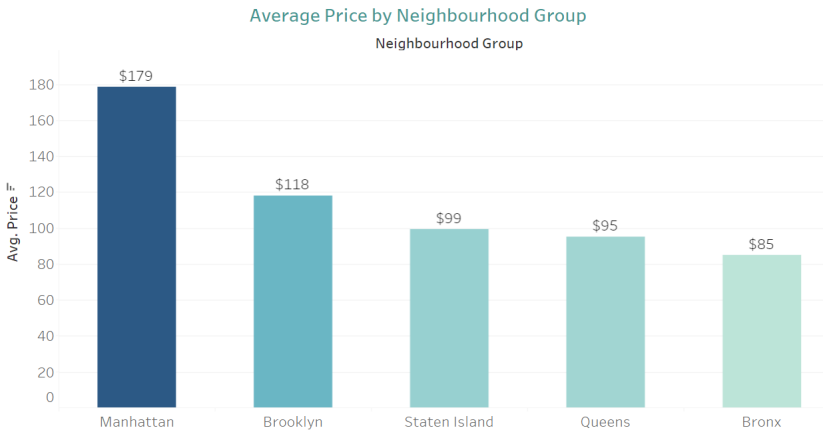
6. Important Findings

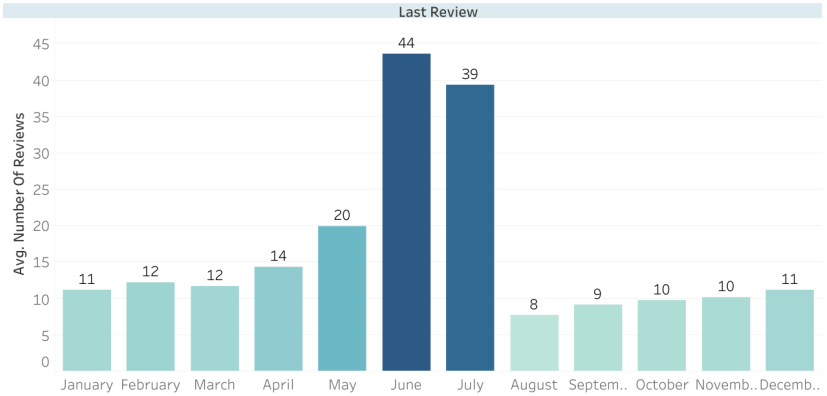


- Focus on Entire Home/Apt: Prioritize expanding offerings in this category (52% preference) to meet demand for private and secluded stays. Emphasize features like privacy, comfort, and exclusive amenities.
- Leverage Private Room Options: Attract guests with personalized and intimate stays (45.7% preference). Highlight unique aspects of each private room, such as cozy ambiance and homely atmosphere.

- Explore Shared Room Potential: Optimize shared rooms (2.4% preference) for budget-conscious or social-oriented travellers. Create safe and comfortable environments, emphasizing cleanliness and privacy partitions.

Average price in the neighbourhoods – room type: All





More Findings

- June received around 0.5 Million Reviews, followed by July which received 0.2 Million reviews.
- 97.6% of listings in Entire Home/Apt and private rooms for less than 10 minimum nights stay.
- Around *80% of total revenue ($37.3M)* is generated from Entire home/apt. So focus should be done on this room type to recover losses.
- Coastal Neighbourhoods: Waterfront / coastal neighbourhoods have a higher concentration of Airbnb listings, offering attractive locations for visitors.
- Manhattan: Most expensive neighbourhood group ($179 avg. price), indicating high demand and popularity among tourists.

7. Tools used

- Python used for Data Understanding, Pre-processing and general Univariate and Bivariate Analysis.
- Tableau & Excel were used for in-depth Bi-Multivariate Analysis.
-

8. Team Members

- Amit Yadav
- K Jagannath
- Pallavi Zadokar
