STORYTELLING CASE STUDY: AIRBNB, NYC

BY

RAGUL | POOJA KUMARI | NEELANJAN ROY

Methodology Document: Presentation I

For Airbnb storytelling case study, we have used Python Jupyter notebook for data cleaning and data processing.

- Importing the dataset: Necessary libraries has been imported and further, dataset has been imported: AB_NYC_2019.csv

```
# importing necessary libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
# importing warnings

import warnings
warnings.filterwarnings('ignore')
```

```
# Reading the dataset

airbnb_df = pd.read_csv("AB_NYC_2019.csv")
airbnb_df.head()              # Checking the head of the dataset
```

- Checked the shape of the dateframe: Dataset has 48,895 rows and 16 columns.
- Checked the overview of the dataframe: We could see that there are 16 columns. Presence of missing values could be seen.

```
# Checking the overview of the dataset

airbnb_df.info()

# We could see that there are 16 columns. Presence of missing values could be seen.
```

```
<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
```

- Checking the statistical summary of the dataframe with describe() function: Presence of outliers could be seen in price column.

```
# Checking the statistical summary of the dataset

airbnb_df.describe([0.5,0.6,0.7,0.8,0.9,0.95,0.99])
```

	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.
50%	1.967728e+07	3.079382e+07	40.723070	-73.955680	106.000000	3.000000	5.000000	0.720000	1.
60%	2.250310e+07	4.868555e+07	40.738420	-73.948890	130.000000	3.000000	9.000000	1.060000	1.
70%	2.714074e+07	8.082452e+07	40.756030	-73.941290	155.000000	4.000000	17.000000	1.690000	2.
80%	3.053027e+07	1.371296e+08	40.770982	-73.927698	200.000000	6.000000	33.000000	2.470000	3.
90%	3.399094e+07	2.128114e+08	40.804890	-73.907810	269.000000	28.000000	70.000000	3.630000	5.
95%	3.525910e+07	2.417646e+08	40.825643	-73.865771	355.000000	30.000000	114.000000	4.640000	15.
99%	3.623888e+07	2.677118e+08	40.864661	-73.776920	799.000000	45.000000	214.000000	7.195800	232.
max	3.648724e+07	2.743213e+08	40.913060	-73.712990	10000.000000	1250.000000	629.000000	58.500000	327.

- Checked the missing values in the dataset: airbnb_df.isna().sum()

```
# Checking the missing values in the dataset

airbnb_df.isna().sum()
```

```
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
```

- Checking the missing values in reviews_per_month column: airbnb_df[airbnb_df['reviews_per_month'].isna()].head()

```
airbnb_df[ airbnb_df['reviews_per_month'].isna()].head()

#Upon checking, missing values in reviews_per_month column is associated with number_of_reviews column. Properties which has not
```

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_revi
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harlem	40.80902	-73.94190	Private room	150	3	
19	7750	Huge 2 BR Upper East Cental Park	17985	Sing	Manhattan	East Harlem	40.79685	-73.94872	Entire home/apt	190	7	
26	8700	Magnifique Suite au N de Manhattan - vue Cloitres	26394	Claude & Sophie	Manhattan	Inwood	40.86754	-73.92639	Private room	80	4	
36	11452	Clean and Quiet in Brooklyn	7355	Vt	Brooklyn	Bedford-Stuyvesant	40.68876	-73.94312	Private room	35	60	
38	11943	Country space in the city	45445	Harriet	Brooklyn	Flatbush	40.63702	-73.96327	Private room	150	1	

```
#Replacing missing values in reviews_per_month with 0

airbnb_df['reviews_per_month'].fillna(0,inplace=True)
```

- Upon checking, missing values in reviews_per_month column is associated with number_of_reviews column. Properties which have not received any reviews has been filled as NaN in reviews per month. Hence, missing values in reviews_per_month is replaced with 0.

- last_review column does not add any value to analysis. Hence, the column has been dropped.

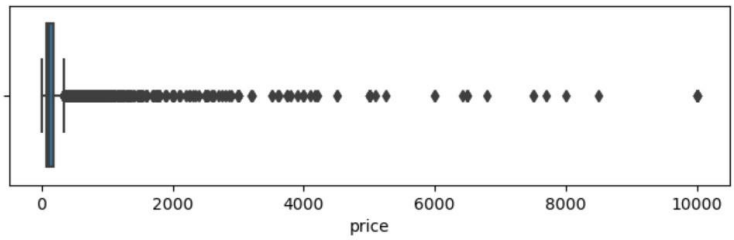
```
#last_review column does not add any value to analysis. Hence, dropping the column

airbnb_df.drop(['last_review'],axis=1,inplace=True)
```

- Checked for outliers in price column: sns.boxplot(data = airbnb_df, x = "price")

```
# Checking for outliers

plt.figure(figsize=[8,2])
sns.boxplot(data = airbnb_df, x = "price")
plt.show()
```



- Spread of prices could be justified with varying amounts of rates fixed by the stay places.
- Checked the distribution of neighbourhood_group: airbnb_df['neighbourhood_group'].value_counts(normalize=True)

```
# Checking the distribution of neighbourhood_group

neigh_dist = (airbnb_df['neighbourhood_group'].value_counts(normalize=True))*100
neigh_dist
```

```
Manhattan        44.301053
Brooklyn         41.116679
Queens           11.588097
Bronx             2.231312
Staten Island     0.762859
Name: neighbourhood_group, dtype: float64
```

- Created a column by binning the price column: pd.cut(x=airbnb_df['price'], bins = [0, 50, 100, 200, 300, 500, 700, 1000, 10000], labels = ['<50', '50-100', '100-200', '200-300', '300-500', '500-700', '700-1000', '1000-10000'])

```
# Created a column by binning the price column

airbnb_df['price_bin'] = pd.cut(x=airbnb_df['price'], bins=[0,50,100,200,300,500,700,1000,10000],labels=['<50','50-100','100-200
airbnb_df[['price','price_bin']].head(10)
```

	price	price_bin
0	149	100-200
1	225	200-300
2	150	100-200
3	89	50-100
4	80	50-100
5	200	100-200
6	60	50-100
7	79	50-100
8	79	50-100
9	150	100-200

- Created a new column by binning the minimum_nights column: pd.cut(x=airbnb_df['minimum_nights'], bins = [0,1,2,3,4,5,10,30,1000], labels=['1N','2Ns','3Ns','4Ns','5Ns','5-10Ns','10-30Ns','>30Ns'])

```
#Binning the minimum nights

airbnb_df['min_nights_bins'] = pd.cut(x=airbnb_df['minimum_nights'], bins=[0,1,2,3,4,5,10,30,1000], labels=['1N','2Ns','3Ns','4Ns
airbnb_df[['minimum_nights','min_nights_bins']].head(10)
```

	minimum_nights	min_nights_bins
0	1	1N
1	1	1N
2	3	3Ns
3	1	1N
4	10	5-10Ns
5	3	3Ns
6	45	>30Ns
7	2	2Ns
8	2	2Ns
9	1	1N

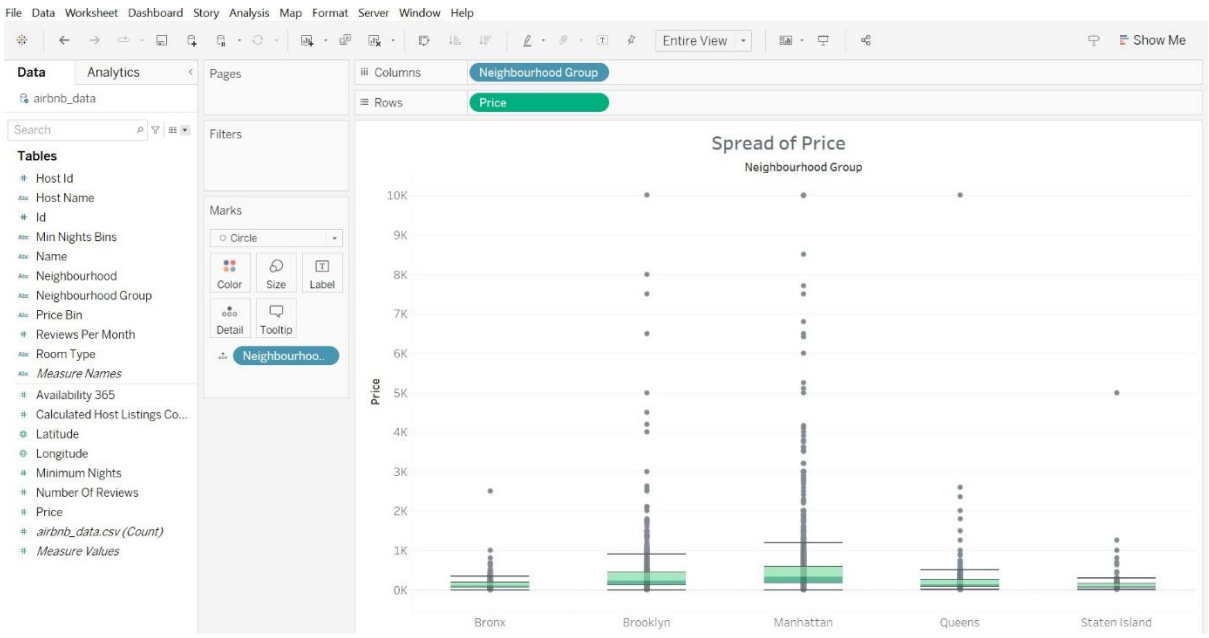
- Saved the updated dataframe to CSV.

```
# Saving the updated dataframe to CSV

airbnb_df.to_csv(r'D:\Data Science\Airbnb, NYC Case Study\airbnb_data.csv', index=False, header=True)
```

Data Analysis & Visualization with Tableau:

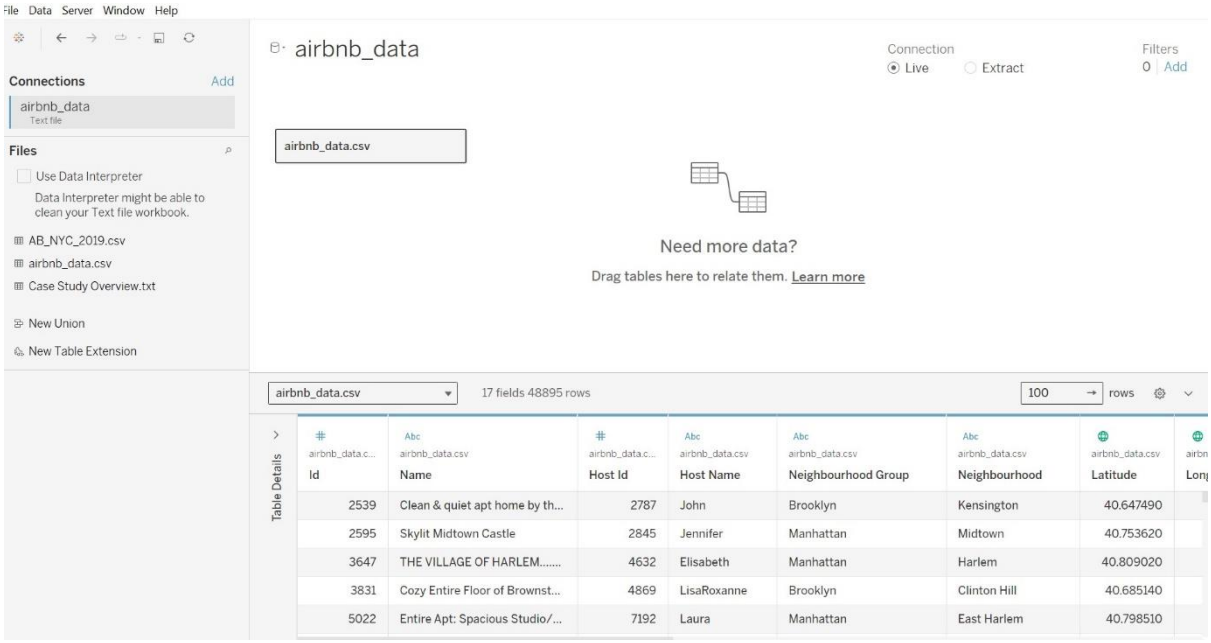
1. Top neighbourhood groups based on bookings:
    - Created a pie chart to check the share of bookings across the neighbourhood groups.
    - Additionally, bar chart has been created to check top neighbourhood based on number of bookings.
2. Bookings based on room types:
    - Created a donut chart with total count of bookings in the centre of the donut and share of bookings among the room types.
    - Created a pie chart for 5 neighbourhood groups to display the share of bookings based on the different room types.
3. Distribution of price among neighbourhood groups:
    - Created a box plot to check the distribution of price across different neighbourhood groups.



4. Price distribution based on room types:
    - Created a bar chart to check the distribution of bookings across different price buckets created. Eg: 50-100, 100-200 etc.,
    - Additionally, bar chart has been created for different room types to visualize the distribution of bookings across different price ranges.
5. Bookings based on minimum nights:
    - Created a bar chart to check the distribution of bookings across the different minimum nights buckets for the different price bins.
6. Availability based on price and minimum nights:
    - Created a bar chart to check the average availability of properties among the different price ranges for the different minimum nights bins.
7. Price & availability based on neighbourhood group:
    - Created a tree maps for different neighbourhood groups and added two dimensions: Size for average price and colour intensity for average availability.
    - Further, created a highlight table for average price across the different neighbourhood group for the different room types.

Methodology Document: Presentation II

CSV file: airbnb_data.csv is connected as data source for visualization tableau



1. Top hosts based on room type:
    - Created a packed bubble chart with hosts based on the number of bookings
    - Additionally, created a bar chart to check the distribution of bookings among the top 5 hosts for different room types.
2. Bookings based on room type:
    - Created a donut chart with total count of bookings in the centre of the donut and share of bookings among the room types.
    - Further, created dual axis chart with bar chart representing the bookings based on different room types and line graph representing the average price based on different room types.
3. Price based on neighbourhood group:
    - Created a pie chart to check the share of bookings across the neighbourhood groups.
    - Further, created a highlight table for average price across the different neighbourhood group for the different room types.
4. Focusing on 50 –200 USD rooms for higher revenue:
    - Created a bar chart to check the distribution of bookings across different price buckets created. Eg: 50-100, 100-200 etc.,
    - Created a bar chart to check the distribution of bookings across different price buckets for the different neighbourhood groups.
5. Availability of minimum one night listings needed to be increased:
    - Created a bar chart to check the distribution of bookings across different minimum nights buckets.

- Created a bar chart to check the average availability of properties across different price buckets for the different minimum nights buckets.

6. Top neighbourhood and availability:
   - Bar chart has been created to check top neighbourhood based on number of bookings.
   - Additionally, a bar chart has been created to check the average price of listings among the different neighbourhoods with dual axis line graph for availability.
7. Availability based on neighbourhood group:
   - Created a tree map for different neighbourhood groups and added two dimensions: Size for average price and colour intensity for average availability.
   - Created a bar chart to check the distribution of bookings across different minimum nights buckets for the different neighbourhood groups.
8. Reviews based on neighbourhood:
   - Created a bar chart to check the distribution of reviews across different neighbourhood with dual axis line graph for average price.
   - Created a bar chart to check the distribution of reviews for price bins.