

ALY 6070:
COMMUNICATION AND VISUALIZATION
FOR DATA ANALYTICS

Final Project: Airbnb New York Analysis
(2011 -2019)

Submitted to:
Prof. Fatemeh Ahmadi Abkenari

Submitted by:

Abhilash Dikshit
Milan Prajapati
Shamim Sherafati
Smit Parmar

Graduate Student at College of Professional Studies in Analytics Northeastern University,
Vancouver, BC, Canada

Master of Professional Studies in Analytics

April 03, 2023

Introduction:

Airbnb has been facilitating unique and tailored travel experiences for both guests and hosts since 2008. The present dataset provides comprehensive information on the metrics and listing activity specific to New York City in 2019.

Dataset:

This dataset contains information about Airbnb host listings, neighbourhoods, price, minimum nights, review counts/rate, longitude, latitude, and the availability throughout the year.

Limitations:

No data on customer stay duration, timeline, review score for each listing and no. of tourist attractions nearby.

Acknowledgements:

Dataset taken from [Kaggle](#). This is a **public dataset of Airbnb**, and the original source can be found on their website.

Dataset Output:

	id <i><dbl></i>	name <i><chr></i>		host_id <i><dbl></i>	host_name <i><chr></i>	neighbourhood_group <i><chr></i>	
1	13157147	Spacious 1 BR Apartment w/ private backyard		2708284	Agustina	Brooklyn	
2	13149690	Private Cozy Bedroom - Central Park North		72008788	Chris	Manhattan	
3	13135822	Gorgeous Bedroom in Manhattan Midtown West		54454582	Breath Beyond	Manhattan	
4888	6809332	Cozy, Warm Home in the West Village		198010	Shenaz	Manhattan	
4889	6657454	Spacious & sunny in Clinton Hill :)		6913285	Marc	Brooklyn	
4890	6638377	Lightfilled studio in Brooklyn		31889552	Chris	Brooklyn	

6 rows | 1-6 of 16 columns

◀	neighbourhood <i><chr></i>	latitude <i><dbl></i>	longitude <i><dbl></i>	room_type <i><chr></i>	price <i><dbl></i>	minimum_nights <i><dbl></i>	number_of_reviews <i><dbl></i>	▶
	Bedford-Stuyvesant	40.69	-73.95	Private room	75	6	37	
	Harlem	40.80	-73.95	Private room	75	3	107	
	Hell's Kitchen	40.76	-73.99	Private room	75	4	146	
	West Village	40.74	-74.01	Entire home/apt	95	5	4	
	Bedford-Stuyvesant	40.69	-73.96	Entire home/apt	95	5	4	
	Fort Greene	40.69	-73.97	Entire home/apt	95	3	1	

6 rows | 7-13 of 16 columns

◀	number_of_reviews <i><dbl></i>	last_review <i><chr></i>	reviews_per_month <i><dbl></i>	calculated_host_listings_count <i><dbl></i>	availability_365 <i><dbl></i>	
	37	2019-06-17	1.53		1	329
	107	2019-06-13	2.86		1	279
	146	2019-07-06	3.89		3	73
	4	2016-02-21	0.09		1	0
	4	2016-06-03	0.08		1	0
	1	2015-07-27	0.02		1	0

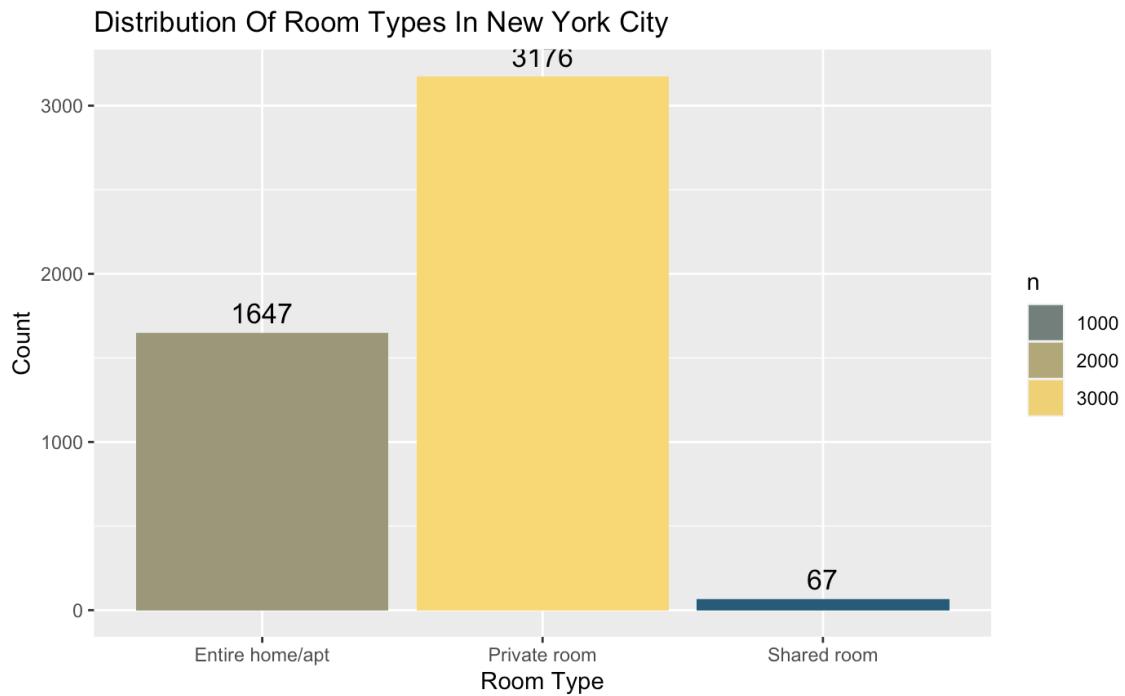
6 rows | 13-17 of 16 columns

Approach 1: R code

1. Bar chart showing the distribution Of Room Types In New York City

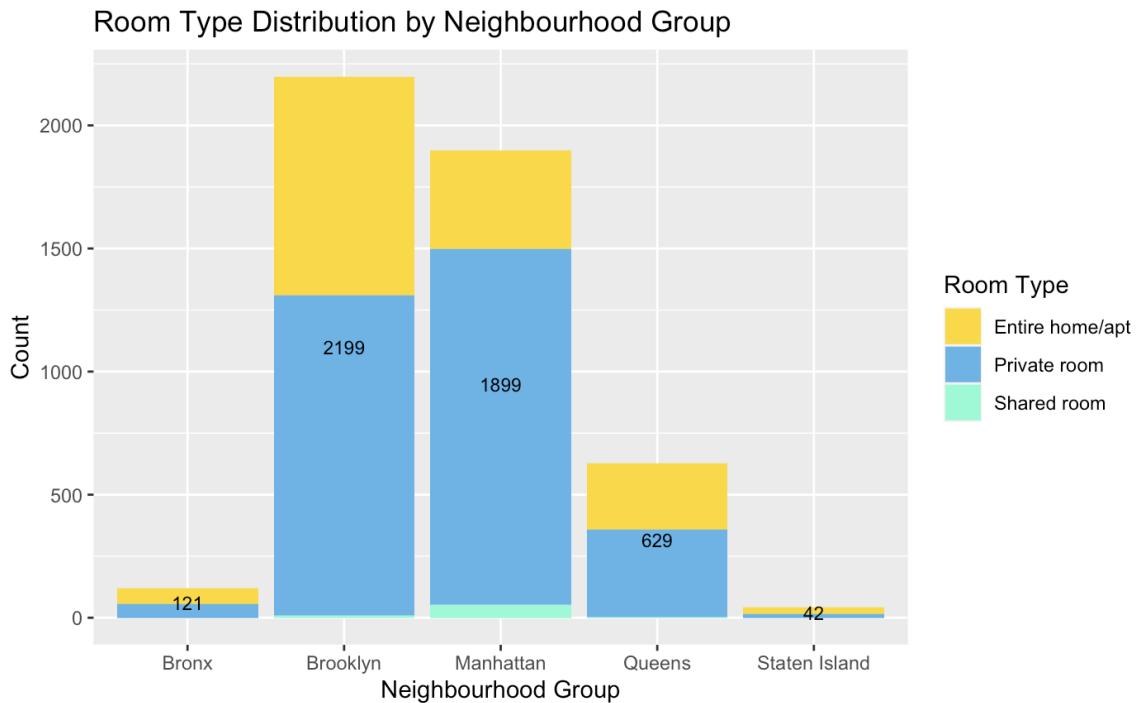
```
counts <- airbnb %>%
  count(room_type)

ggplot(data = counts) +
  geom_bar(mapping = aes(x = room_type, y = n, fill = n),
           stat = "identity", position = "dodge") +
  scale_fill_gradient(low = "#005E7D", high = "#FFD662", guide = "legend") +
  geom_text(aes(x = room_type, y = n, label = n),
            position = position_dodge(width = 0.9),
            vjust = -0.5, size = 4.5) +
  labs(title = "Distribution Of Room Types In New York City",
       x = "Room Type", y = "Count")
```

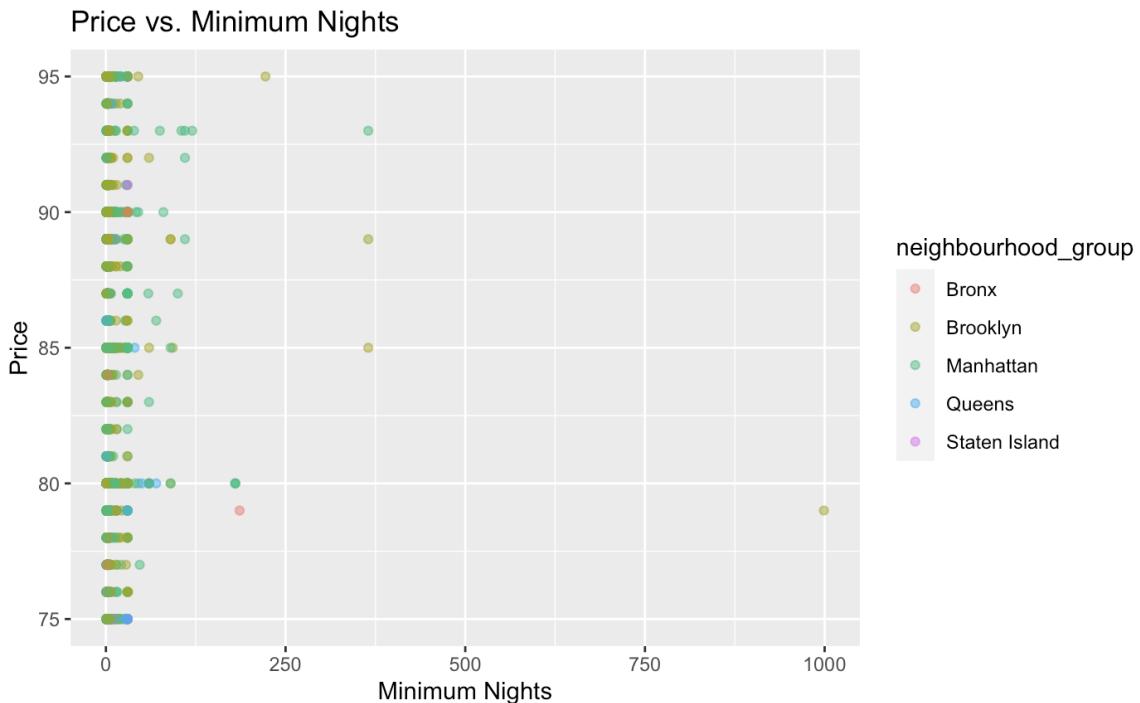


2. Stacked bar chart showing the distribution of room types by neighbourhood group

```
ggplot(data = airbnb) +
  geom_bar(mapping = aes(x = neighbourhood_group, fill = room_type),
           position = "stack") +
  stat_count(mapping = aes(x = neighbourhood_group, label = stat(count)),
             geom = "text", position = position_stack(vjust = 0.5), size = 3) +
  scale_fill_manual(values = c("#FFD700", "#56B4E9", "#7FFFAD")) +
  labs(title = "Room Type Distribution by Neighbourhood Group",
       x = "Neighbourhood Group", y = "Count") +
  guides(fill = guide_legend(title = "Room Type"))
```

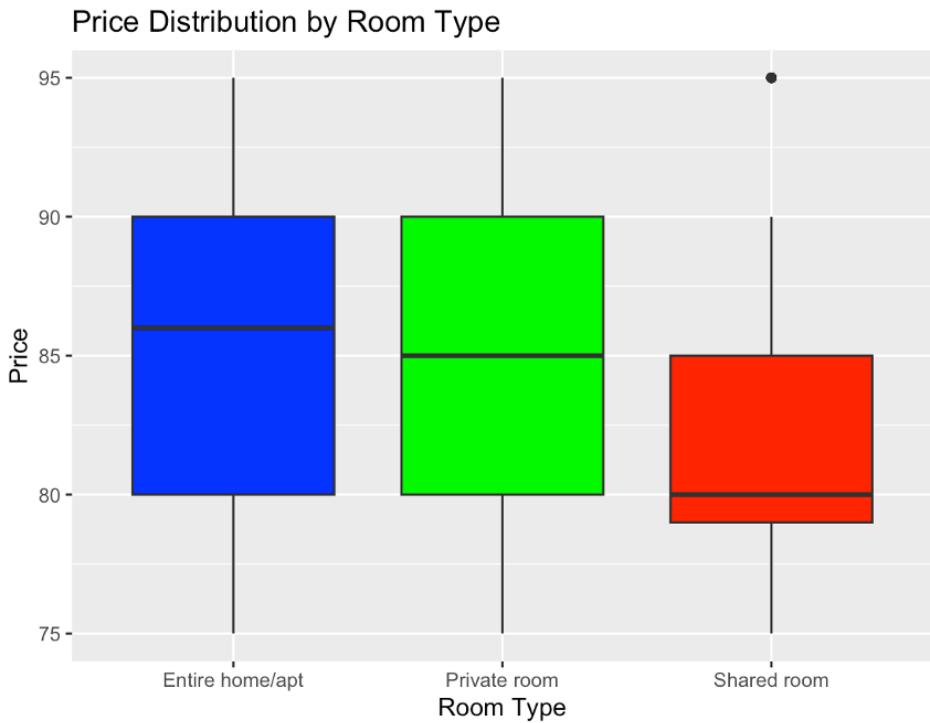


3. Scatterplot showing the relationship between price and minimum nights:



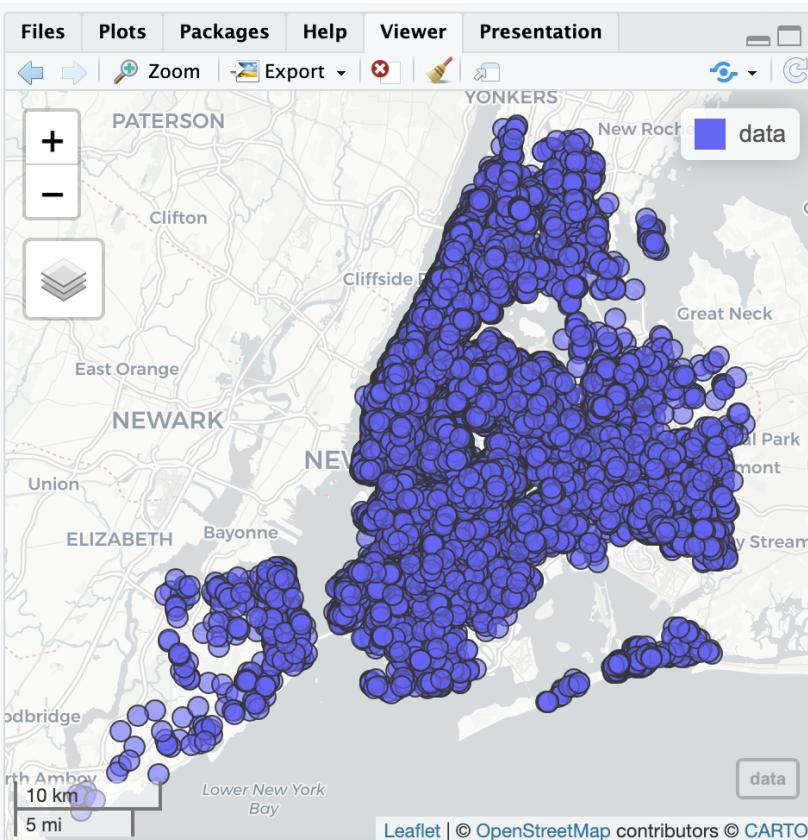
4. Boxplot showing the distribution of prices by room type:

```
ggplot(data = airbnb) +
  geom_boxplot(mapping = aes(x = room_type, y = price, fill = room_type)) +
  scale_fill_manual(values = c("blue", "green", "red")) +
  labs(title = "Price Distribution by Room Type", x = "Room Type", y = "Price")
```

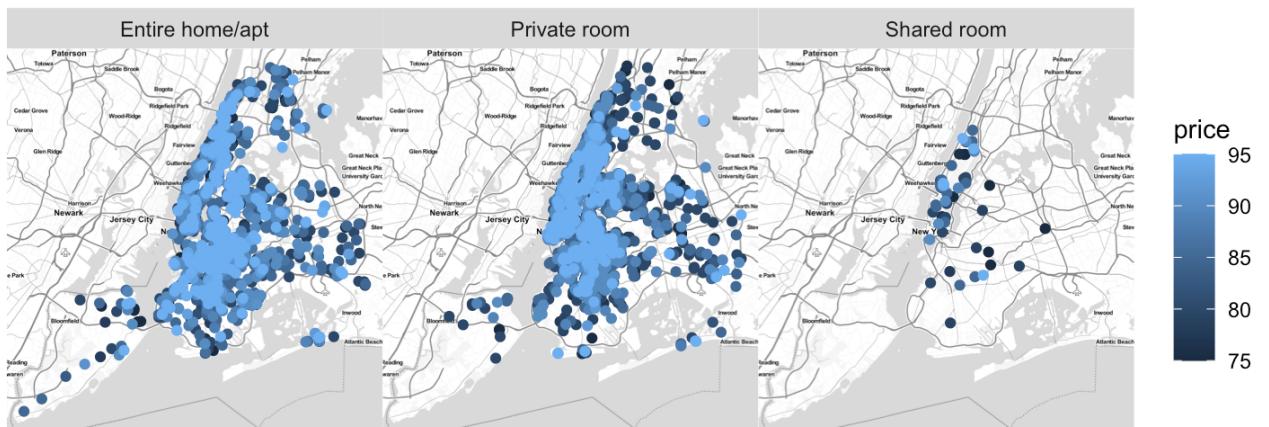


5. Mapview for number of reviews for specific room types as per price (Using Latitude and Longitude)

```
mapview(airbnb_unclean, xcol = "longitude", ycol = "latitude", crs = 4326, grid = FALSE) # coordinate reference system (crs)
```

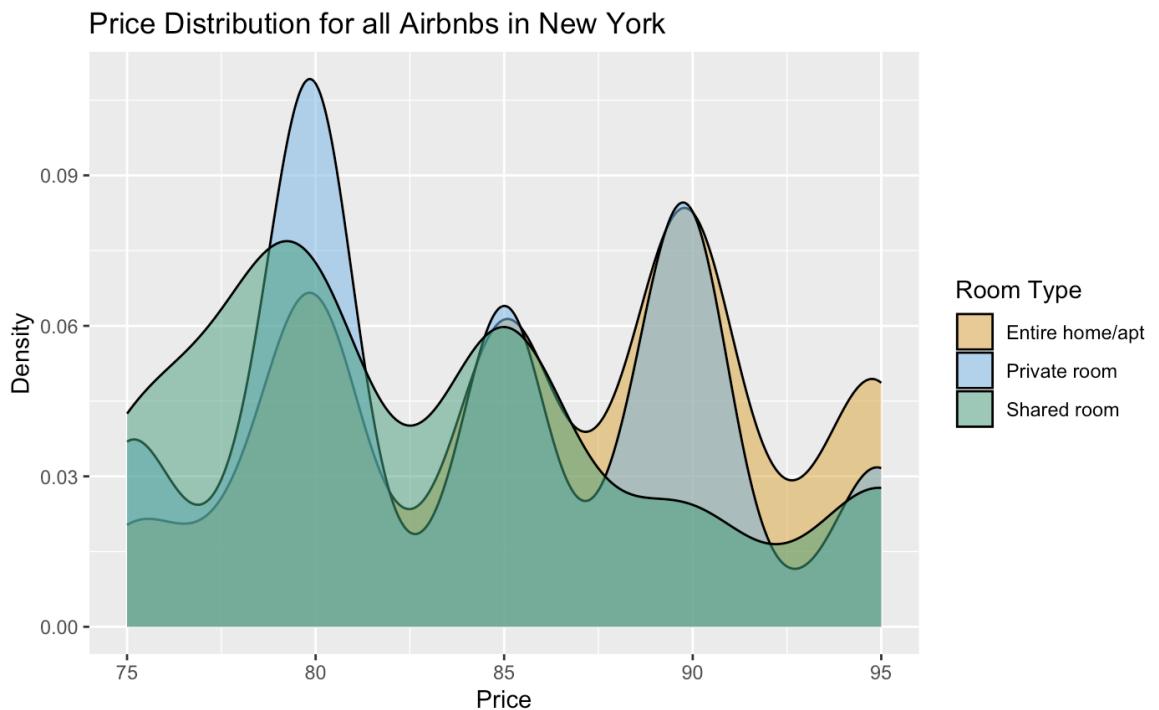


```
qmplot(longitude, latitude, data = airbnb, geom = "point", color = price, main = "Density of Latitude and Longitude") + facet_wrap(~ room_type)
```



6. Density plot showing the distribution of prices:

```
ggplot(data = airbnb, aes(x = price, fill = room_type)) +
  geom_density(alpha = 0.5) +
  scale_fill_manual(values = c("#E69F00", "#56B4E9", "#009E73")) +
  labs(title = "Price Distribution for all Airbnbs in New York", x = "Price", y = "Density", fill = "Room Type")
```

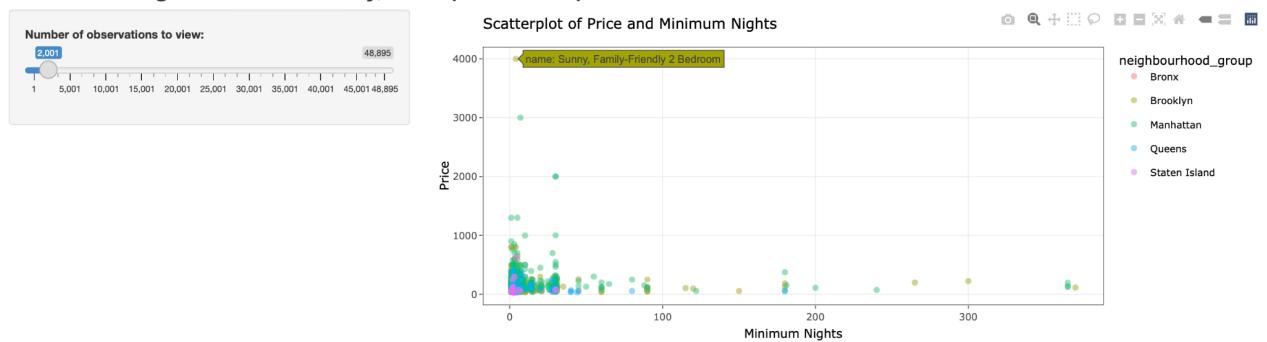


Approach 2: Shiny R

Now, we can create our Shiny app UI and server functions.

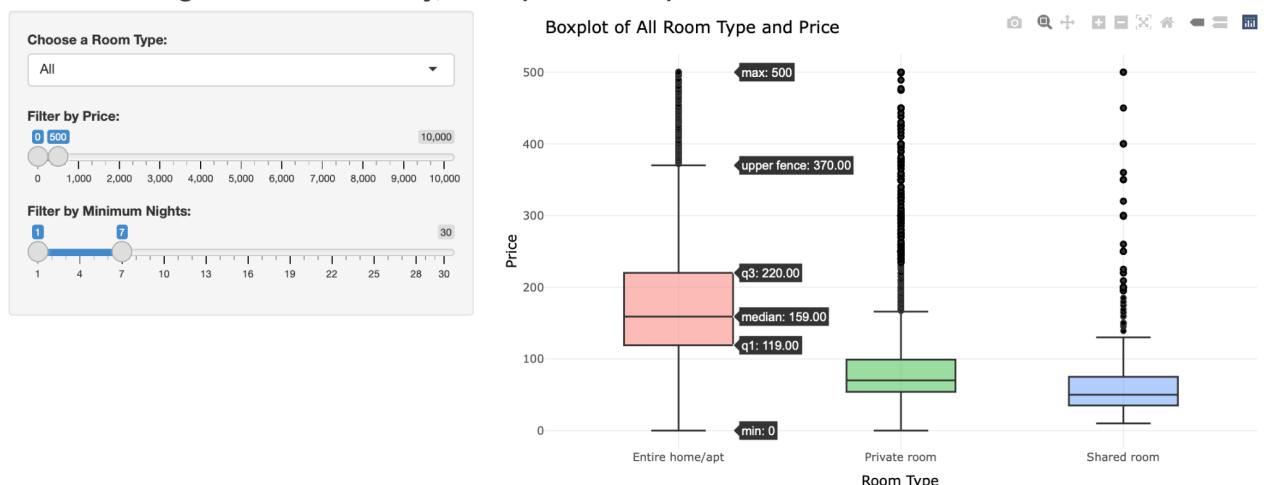
1. Scatter Plot:

Airbnb Listings in New York City, USA (2011-2019)



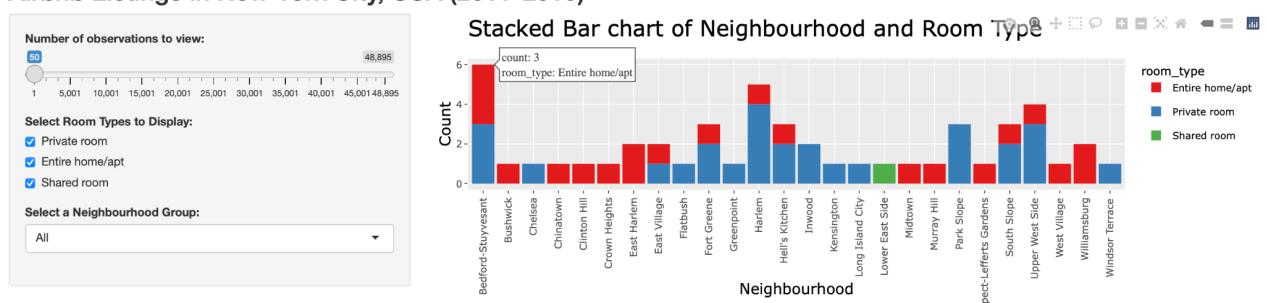
2. Boxplot:

Airbnb Listings in New York City, USA (2011-2019)



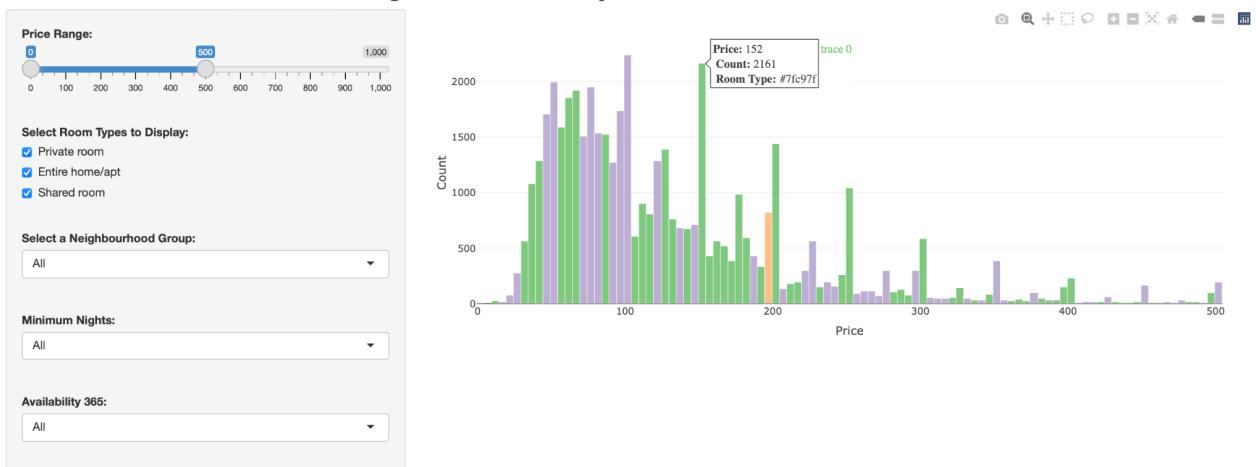
3. Stacked bar chart:

Airbnb Listings in New York City, USA (2011-2019)



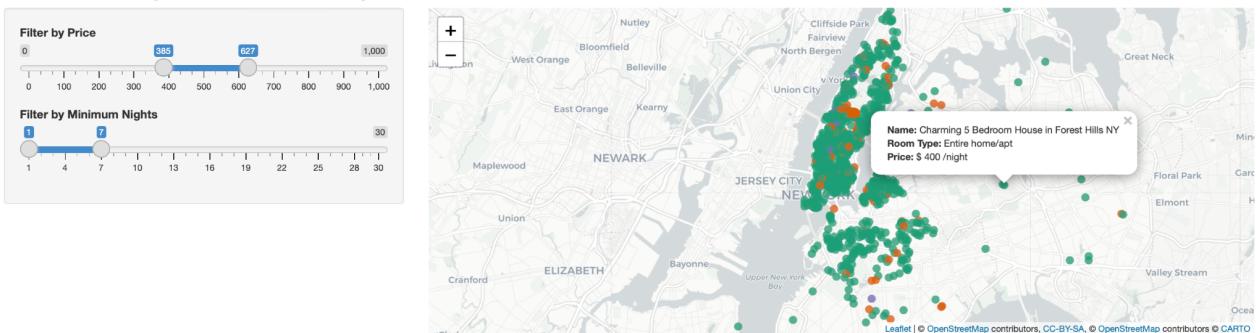
4. Histogram plot:

Price Distribution of Airbnb Listings in New York City



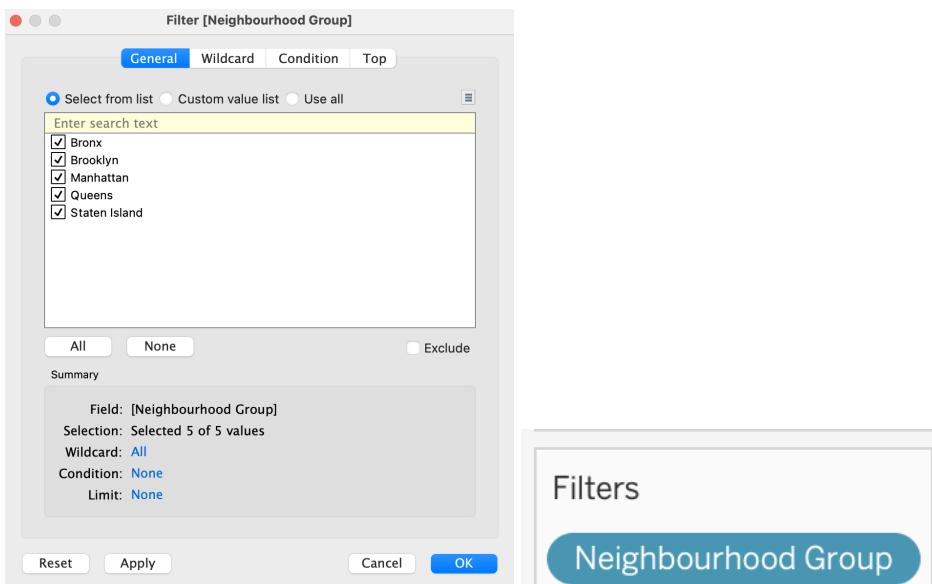
5. Mapview:

Airbnb Listings in New York City

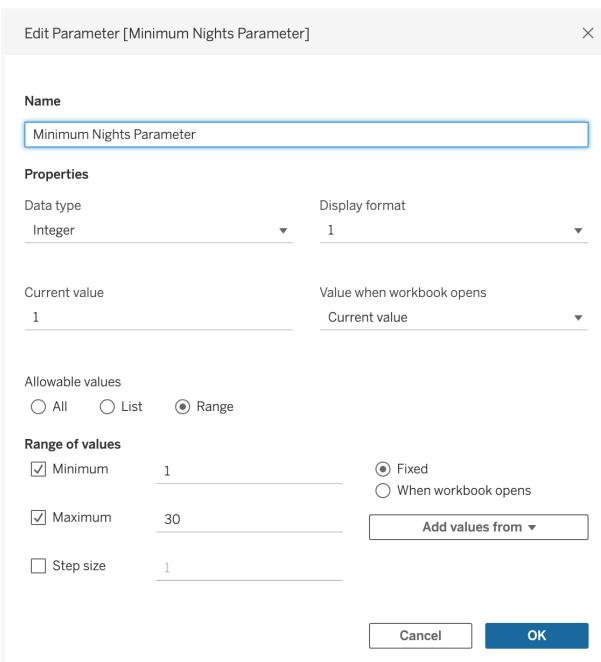


Approach 3: Tableau

1. **Filter:** We created a filter to display only the data for a specific neighborhood. To do this, we added a "neighbourhood_group" field to the Filters shelf, selected the neighborhoods to display. We then used the filter to dynamically change the data displayed in the visualization.



2. **Parameter:** We created a parameter on the “minimum_nights” field to allow the user to change the minimum number of nights for a stay.



We can now use the parameter to dynamically change the data displayed in the visualization.

3. **Dynamic measure:** We created a dynamic measure to show the average price per night on the "price" field and enter the below formula:

$\text{AVG}(\text{IIF}([\text{minimum_nights}] \leq [\text{Minimum Nights Parameter}], [\text{price}], \text{NULL}))$

In the above formula, [Minimum Nights Parameter] refers to the parameter that we have created to allow the user to change the minimum number of nights for a stay.

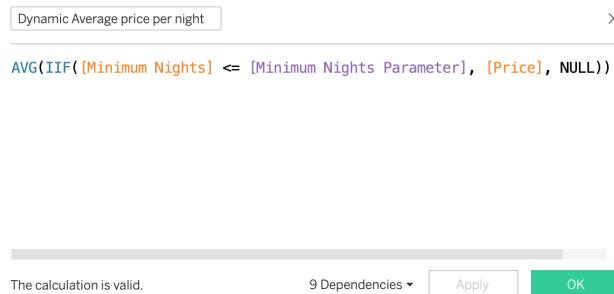
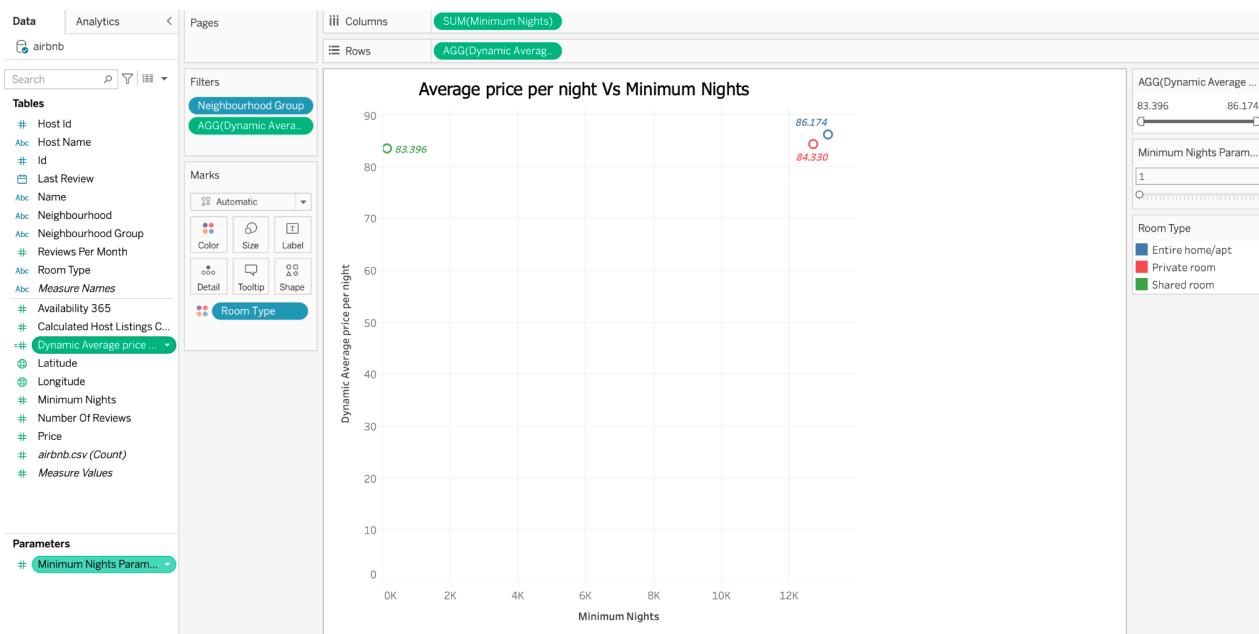
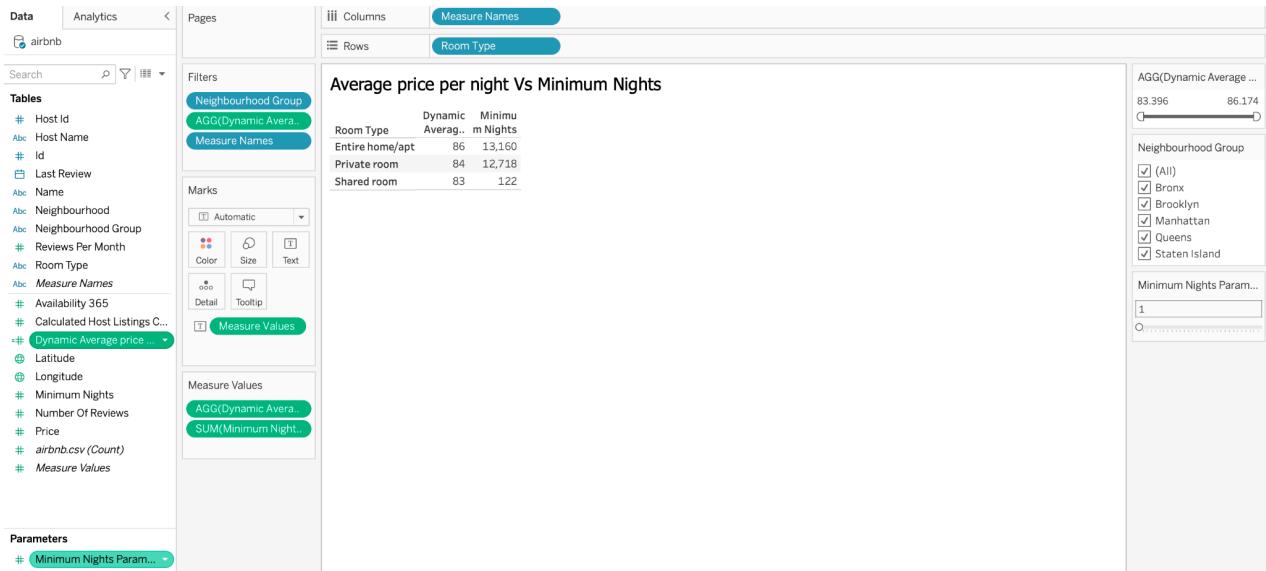


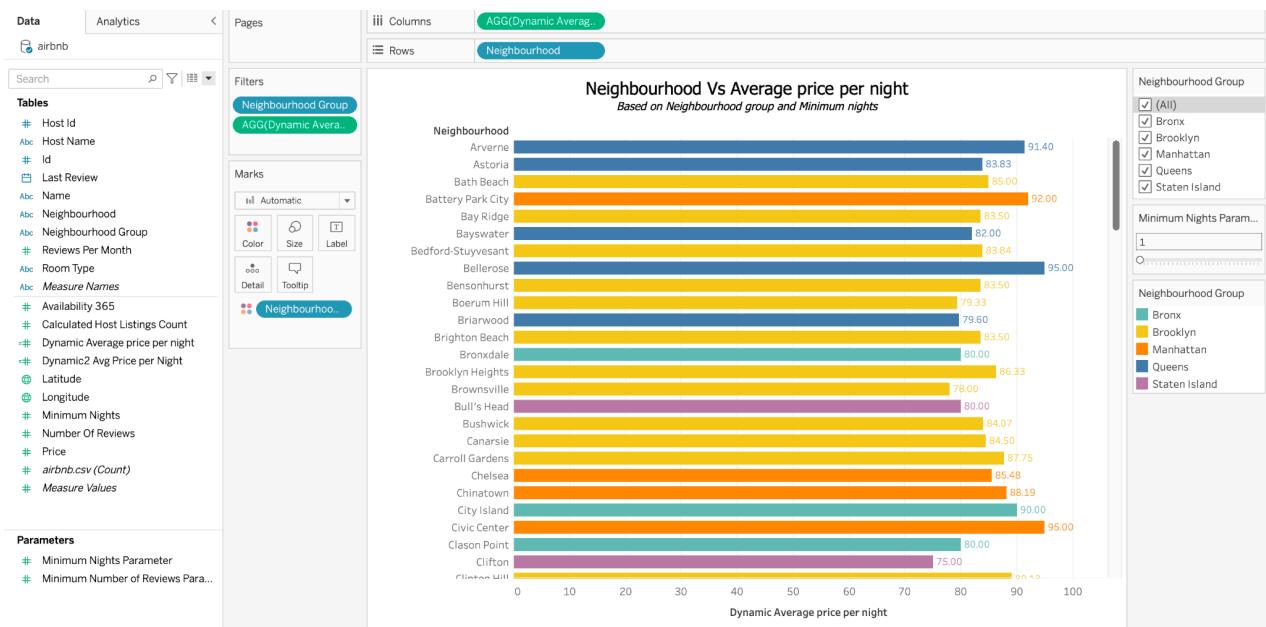
Tableau Visualization based on the filter, parameter, and dynamic measure:

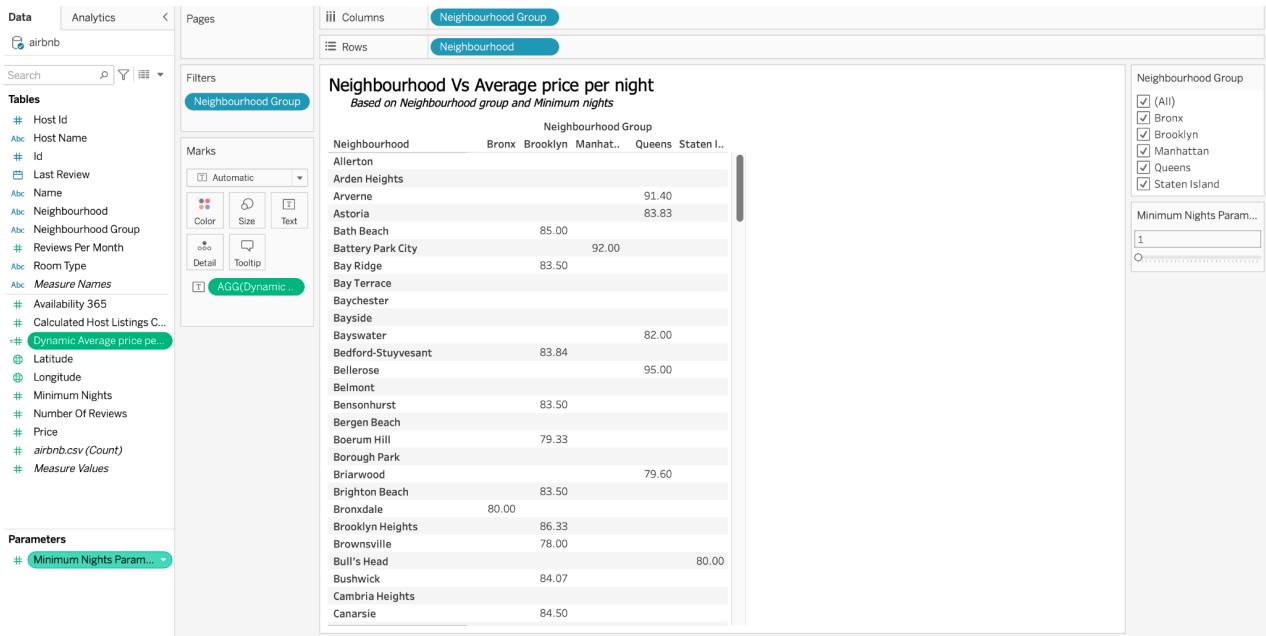
1. **Scatter Plot:** Drag the "minimum_nights" field to the Columns shelf, and the dynamic measure created in the previous step (e.g., "Average Price per Night") to the Rows shelf. We added the "room_type" field to the Color shelf to differentiate between the different types of rooms.



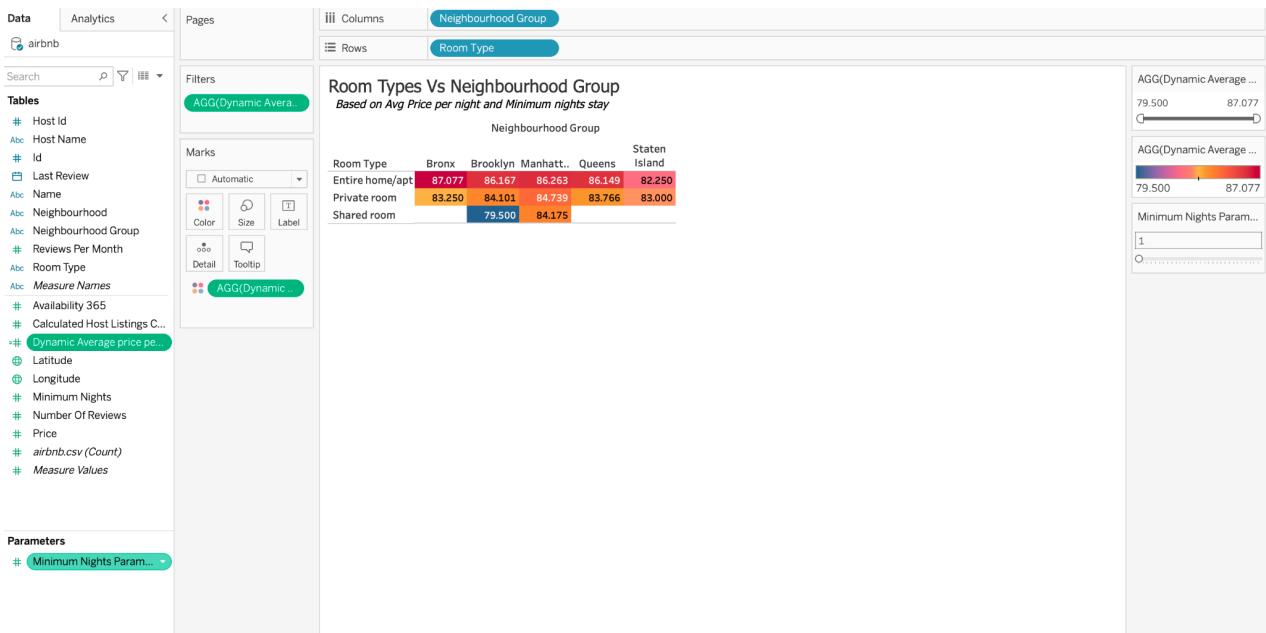


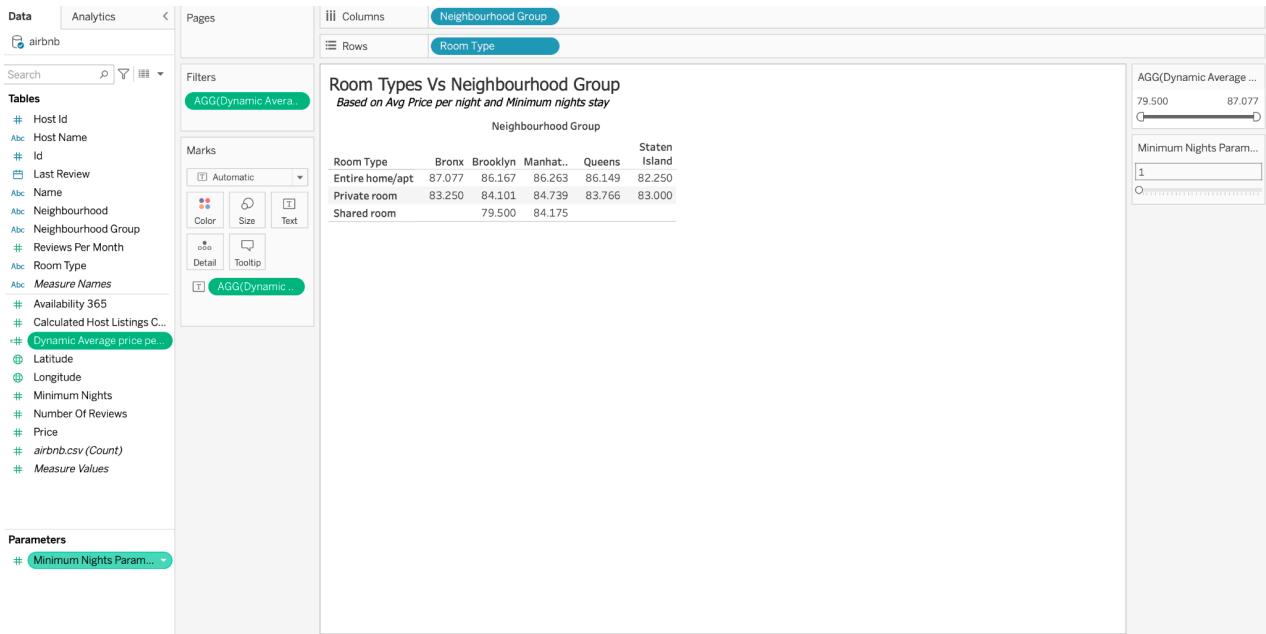
2. **Bar Chart:** Drag the "neighbourhood" field to the Rows shelf, the dynamic measure created in the previous step to the Columns shelf, and the filter created in the first step (e.g., "Neighbourhood Group") to the Filters shelf. This will create a bar chart showing the average price per night for each neighbourhood.



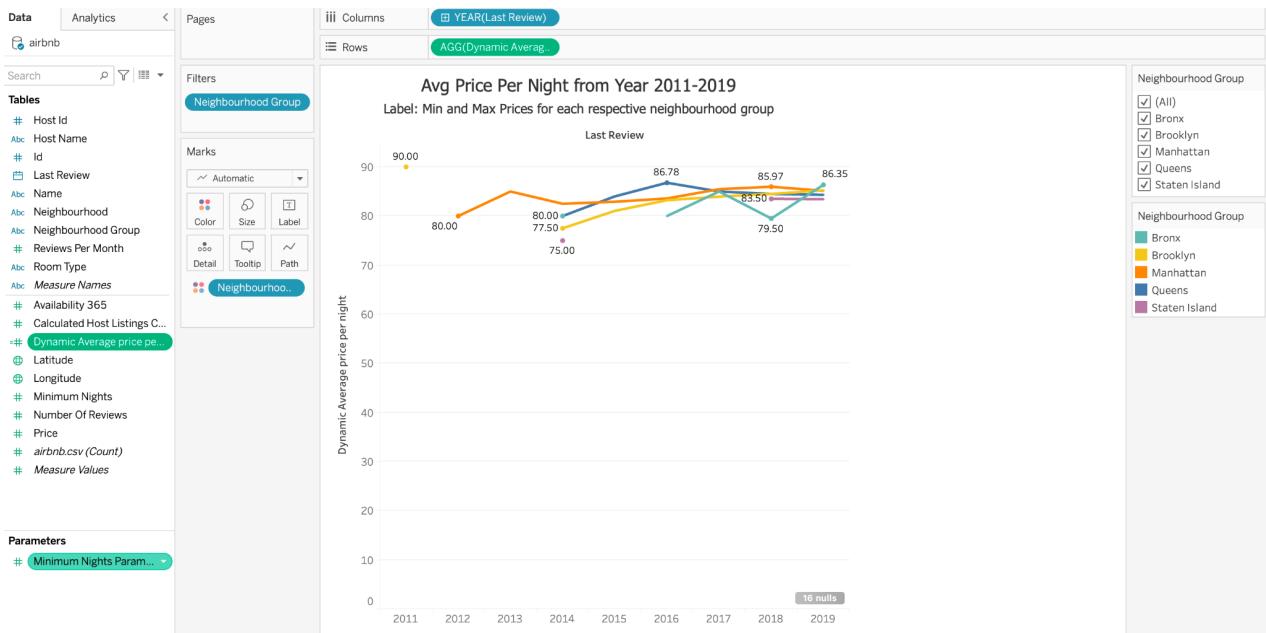


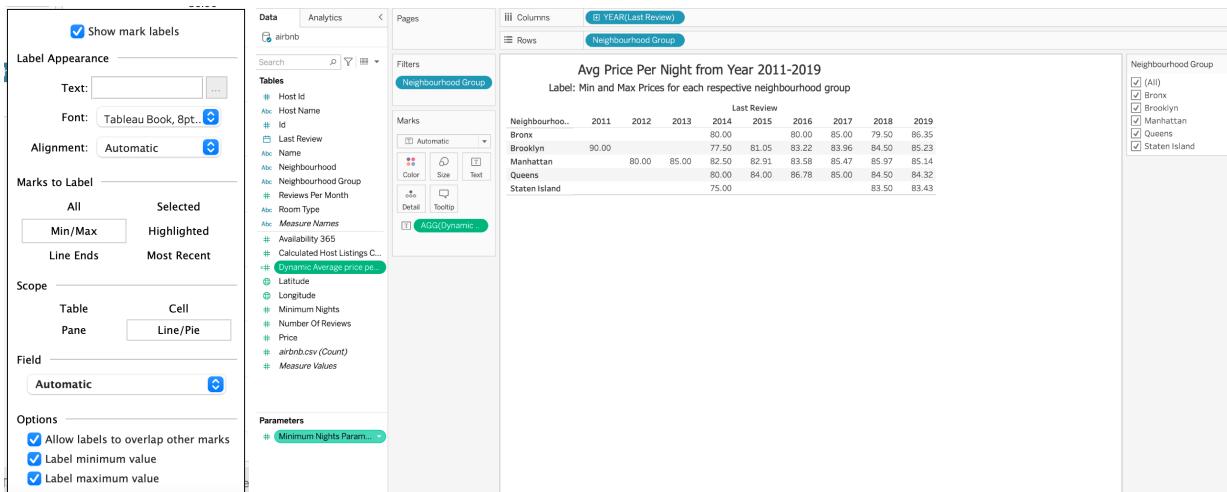
3. **Heat Map:** Drag the "neighbourhood_group" field to the Columns shelf, the "room_type" field to the Rows shelf, and the dynamic measure created in the previous step to the Color shelf. We also added the "minimum_nights" parameter to the Filters shelf, so the user can adjust the minimum number of nights to see how the price per night changes.



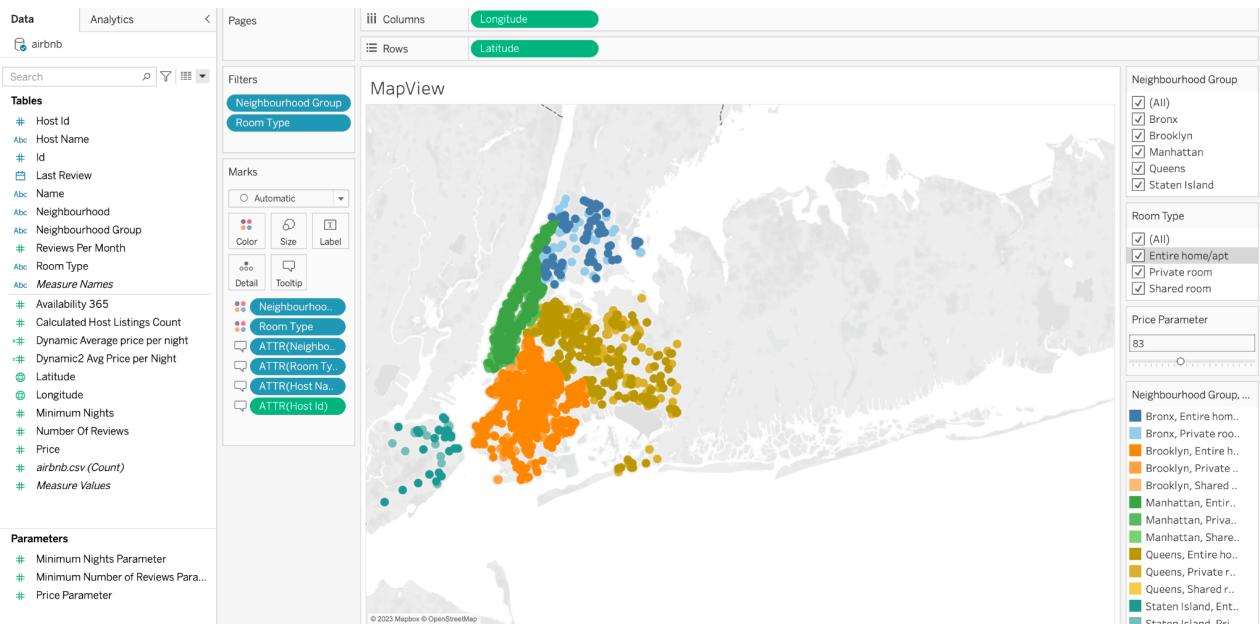


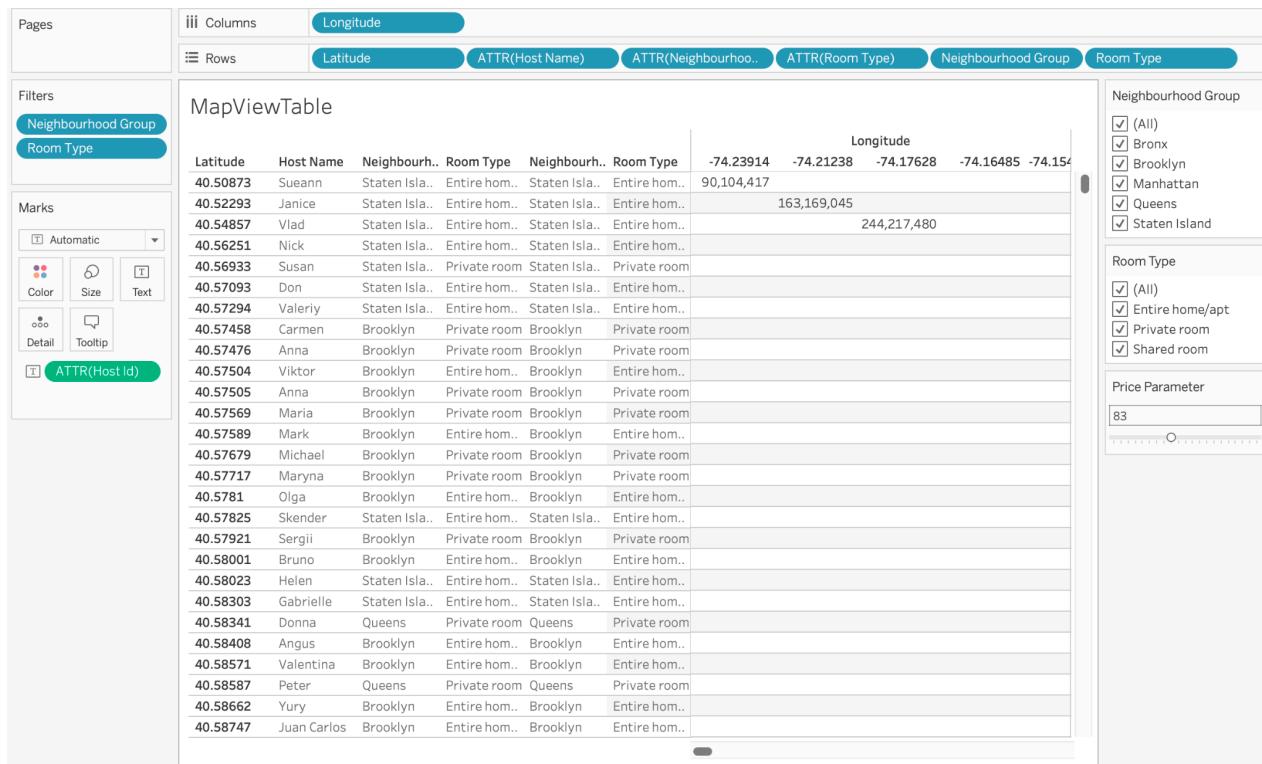
4. **Line Chart:** Drag the "last_review" field to the Columns shelf, the dynamic measure created in the previous step to the Rows shelf, and the "neighbourhood_group" field to the Color shelf. We also added a filter to show data for a specific neighbourhood.





5. Map View: Drag the "host_id" field to the Columns shelf, the "minimum_nights" parameter to the Filters shelf, and the dynamic measure created in the previous step to the Rows shelf. We also added the "room_type" field to the Color shelf to differentiate between the different types of rooms. This chart will show the average price per night for each host, stacked by the type of room.





Conclusion:

The choice of data visualization tool depends on the specific needs and goals of the project. In the case of the dataset provided, R code, Shiny R, and Tableau can all be effective tools for communicating and visualizing the data.

R code is a popular choice for data analysis and visualization because it provides a wide range of customizable graphs and charts. With R, users can create complex visualizations that display multiple data variables, and can use advanced statistical techniques to explore relationships between variables. R code also allows for reproducibility and sharing of analysis and visualizations with others.

Shiny R is a web application framework for R that allows users to build interactive web applications that display data visualizations. Shiny R can be a great choice for projects where stakeholders need to interact with the data and make decisions based on the analysis. Shiny R applications can also be easily customized and updated as needed, allowing for more flexibility in data exploration.

Tableau is a popular data visualization tool that allows users to create interactive dashboards and visualizations. Tableau provides a user-friendly interface that allows users to quickly create charts and graphs with drag-and-drop functionality. Tableau also provides advanced analytics and mapping capabilities, making it a good choice for projects that require more complex visualizations.

In conclusion, all three tools can be effective for visualizing the provided dataset. R code provides more customization and advanced statistical techniques, while Shiny R allows for more interactivity and flexibility in data exploration. Tableau provides

a user-friendly interface and advanced analytics capabilities. Ultimately, the choice of tool will depend on the specific needs and goals of the project, as well as the user's level of experience and comfort with the tool.

Reference:

1. Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2020). shiny: Web Application Framework for R. R package version 1.6.0. Retrieved from <https://CRAN.R-project.org/package=shiny>
2. RStudio. (n.d.). Leaflet for R. Retrieved March 19, 2023, from <https://rstudio.github.io/leaflet/>
3. Kaggle. (2019). Airbnb NYC 2019. Retrieved from <https://www.kaggle.com/dgomonov/new-york-city-airbnb-open-data>
4. R Core Team. (2021). R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from <https://www.R-project.org/>