Introduction

Overview of Airbnb and its User Interface

For the purpose of introducing the firm, Airbnb is a software solution for tourists/users around the world to cite and book accommodations from all over the globe. Airbnb provides a marketplace for a myriad of not only accommodations, but also events such as *experiences and restaurants* These accommodations are not owned by Airbnb, but are privately owned by people. Accommodations include houses, apartments, condominiums, hostels, and hotel rooms.

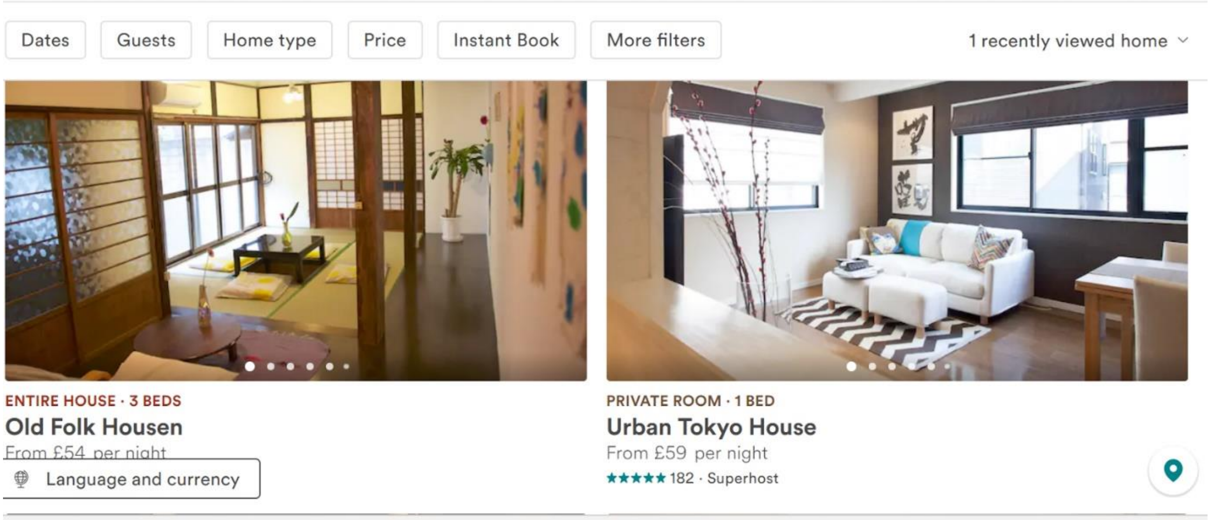
On the company's booking system user interface contains the variety of accommodations to look upon. To help assist customers with the right preferences, Airbnb provides filters/drop-downs. Such filters include number of guests, home type, and price. Other filters include different type of amenities, pet-friendly option, and available facilities.

The Problem of the existing user interface

The Current system lacks information on traveling costs and traveling information. Traveling information can provide with different types of transportation that the customer can take to reach their booked accommodation. For example, tourists could find themselves selecting the filter of traveling by Uber from the airport. Now, the absence of this service can be a concern for tourists who care about expenses and are living on a limited budget.

Overview of Airbnb's UI

Accommodation Navigation:



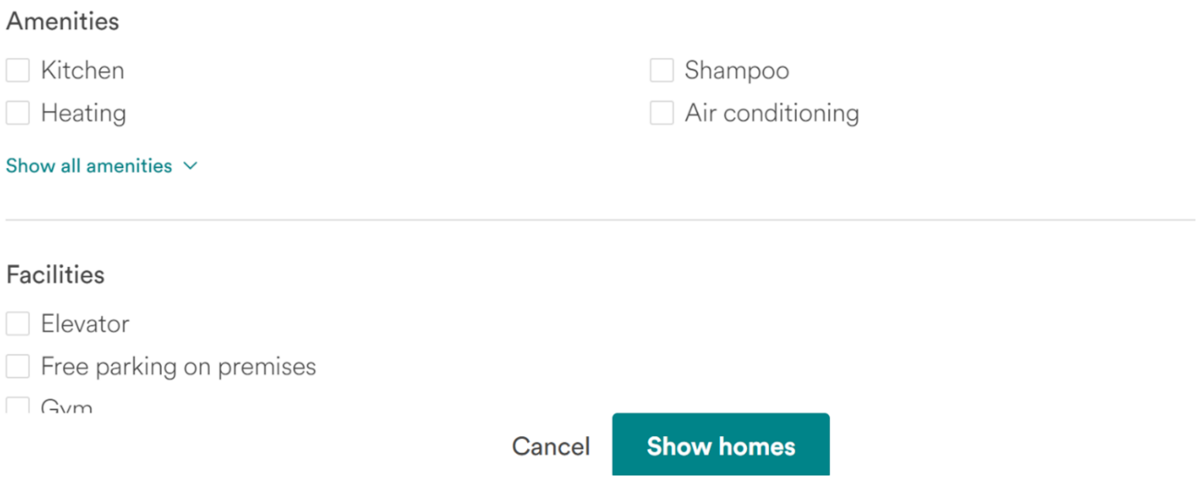
ENTIRE HOUSE · 3 BEDS
Old Folk Housen
From £54 per night

PRIVATE ROOM · 1 BED
Urban Tokyo House
From £59 per night
★★★★★ 182 · Superhost

Description: Type of accommodation, number of beds, name of accommodation, and price

There are also filters like dates, guest, and home type for users to choose.

Other Special Filters:



Amenities

□ Kitchen
□ Heating
□ Shampoo
□ Air conditioning

Facilities

□ Elevator
□ Free parking on premises
□ Gym

Project Proposal

I propose a tool that will be a solution for the problem above. The useful features are listed below:

- The tool will help users browse through accommodations and observe the average Uber fare from the country's main airport to the designated accommodation
- The tool will calculate the total cost information which is calculated with the formula – (2 * Uber fare + Price of accommodation)
- Some of the common filters will be available – (price, name of accommodation, neighborhood, room type)
- Avg Price per neighborhood
- Map showing different accommodations, which can enhance user experience

Programs that will be used for development

- Tableau Public – for core analytics and representation of the tool. The program can be used to create dashboards which can contain many visual graphs and map which the developer can design
- Jupyter Notebook – this software can be used for Python computation The Jupyter notebook can provide a better experience than using the traditional Python IDE since the former contains a notebook that automatically saves your progress. The program is used specially for Data Wrangling and polishing the appropriate data file to use in Tableau.

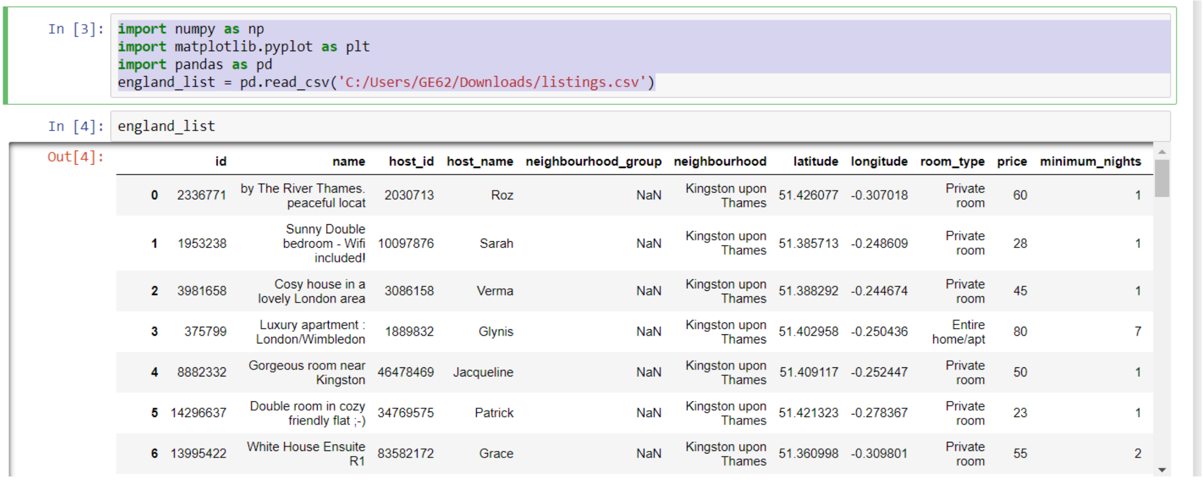
Data source

The raw data that we will be using is from Insideairbnb.com. The website provides users with various data files categorized by various popular cities. The file will contain if not all, most of the accommodations available in governing city. For each Comma-separated value file (.csv file), there will be information on the following variables:

- Name of accommodation
- Host_id
- Host_name

- Neighborhood
- Latitude
- Longitude
- Room_type
- Price (per night)
- Minimum nights

For the project, London data will be used. London contains a variety of neighborhoods.



```
In [3]: import numpy as np
        import matplotlib.pyplot as plt
        import pandas as pd
        england_list = pd.read_csv('C:/Users/GE62/Downloads/listings.csv')

In [4]: england_list
Out[4]:
```

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights
0	2336771	by The River Thames. peaceful locat	2030713	Roz	NaN	Kingston upon Thames	51.426077	-0.307018	Private room	60	1
1	1953238	Sunny Double bedroom - Wifi included!	10097876	Sarah	NaN	Kingston upon Thames	51.385713	-0.248609	Private room	28	1
2	3981658	Cosy house in a lovely London area	3086158	Verma	NaN	Kingston upon Thames	51.388292	-0.244674	Private room	45	1
3	375799	Luxury apartment : London/Wimbledon	1889832	Glynis	NaN	Kingston upon Thames	51.402958	-0.250436	Entire home/apt	80	7
4	8882332	Gorgeous room near Kingston	46478469	Jacqueline	NaN	Kingston upon Thames	51.409117	-0.252447	Private room	50	1
5	14296637	Double room in cozy friendly flat ;-)	34769575	Patrick	NaN	Kingston upon Thames	51.421323	-0.278367	Private room	23	1
6	13995422	White House Ensuite R1	83582172	Grace	NaN	Kingston upon Thames	51.360998	-0.309801	Private room	55	2

Development

Data Wrangling using Jupyter Notebook

Aim and Objectives:

- Find missing values
- Run Exploratory Data Analysis
- Add data parameters – Distance (from airport), Uber fare, Total cost

Determine Missing Values

In determining missing parameters, the function isnull() is used to return a Boolean. I used the function to run on the pandas Series of neighborhood, latitude, longitude, room_type, and price, which are the crucial variables. The results

Exploratory Data Analysis (EDA)

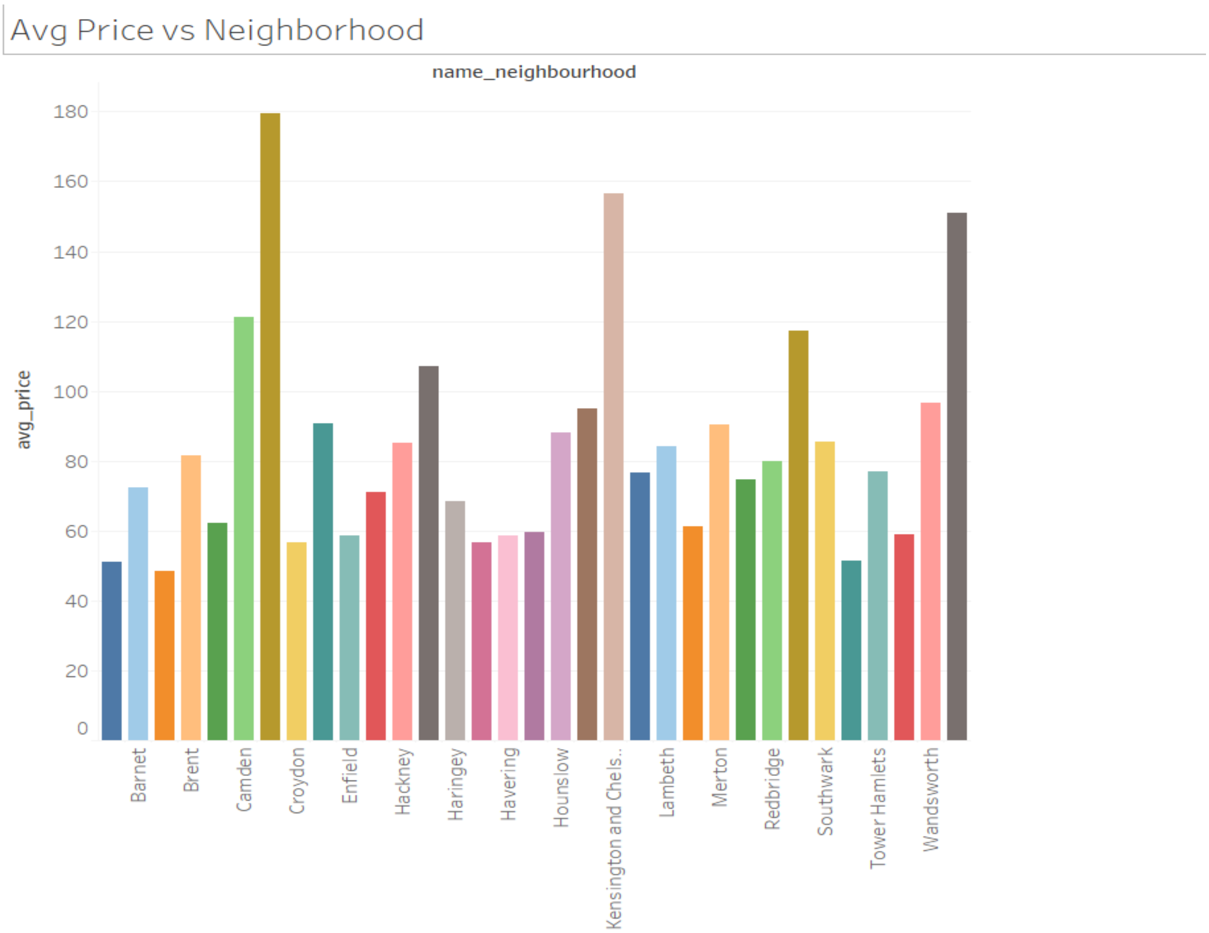
The purpose of running an EDA on data is to have a chance to practice the skills learned from the interactive courses. Running an EDA will provide developer with a visualization that conveys many useful insights that are not available just by using the raw data.

In this EDA, the aim is to use raw data to wrangle and extract existing measures to produce a visualization for various representations.

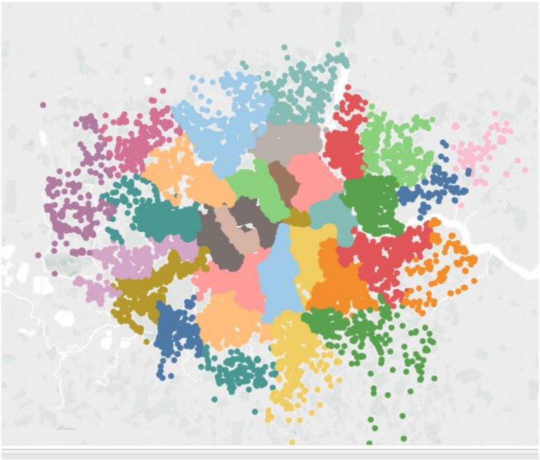
The visualization is done doing Tableau public.

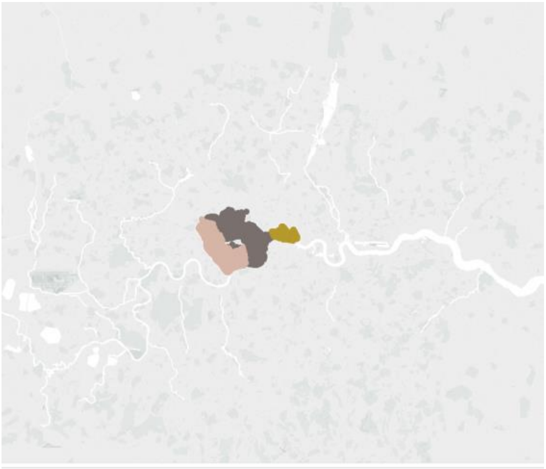
Process

EDA Deliverable: Bar graph (neighborhoods vs average price) created with Tableau Public



This analysis is performed to illustrate how price varies upon different neighborhoods in London. Sub cities in England are called neighborhoods. The graph shows that the highest price of accommodations exists in City of London and followed by Kensington and Chelsea, and Westminster (label not showed on figure). One can easily see that the price of City of London accommodations clearly stood out among others. This suggests that the price of land at The City of London and Kensington/Chelsea is highly valuable and the demand is high. Further analysis can be done to see why these areas are in high demand.

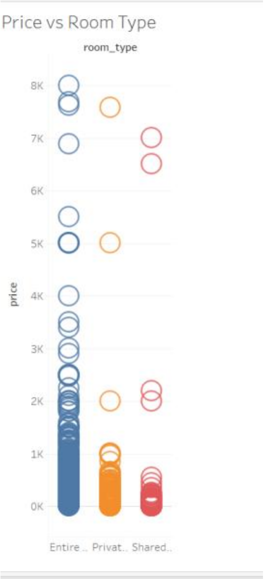




Plotting our accommodations on the map using Tableau, you can see that City of London, Kensington and Chelsea, and Westminster are the center of London city. The case here (in London) seems to be that the more centralized the neighborhood is, the higher the price. Accommodation prices of neighborhood at boundaries surrounding the city such as Havering and Bexley are cheap compared to center neighborhoods.

This representation can give me and consumers the idea that if they want to find cheap places in London, they should try to allocate the three centralized neighborhoods and should to go for the more center distant neighborhoods.

EDA Deliverable: Prices vs Type of Accommodation



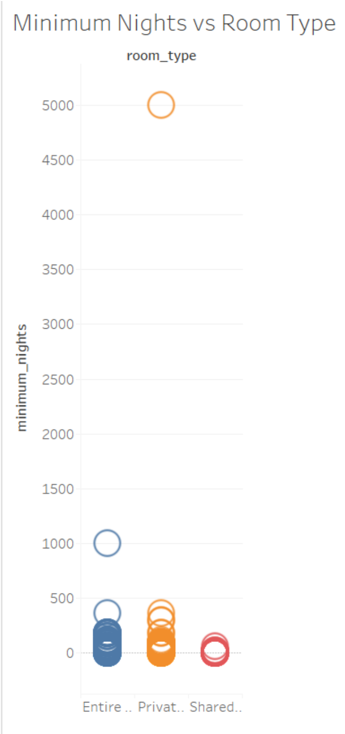
Note: One circle is equivalent to one accommodation. This means that the bolder the visualization, the more populated the distribution is.

This analysis is performed to see if the price varies for the type of rooms. For all types of rooms, one can see that most prices consists within the 2k range. Remember that these prices are based on starting prices for a certain accommodation. A private room can cost 500 for a night while an entire apartment can cost 5000 for 10 nights. Nevertheless, our analysis is valid because a consumer still needs to pay 5000 minimum for the entire apartment.

The visualization clearly depicts that an entire apartment booking will generally cost more than a private room and a private room generally cost more than a shared room. A shared room should generally be cheap because these types of rooms exist in hostels which are uniformly considered to be low cost shelter for tourists. An entire apartment will cost more because you are booking more space for more guests compared to a private room which means allocating more land space.

This shows that the reason the average prices are so high can be due to apartment prices' influence. Out of all accommodation listings, number of apartment listings are highest compared to other room types. This analysis is valuable because one can see that there's a correlation between price and type of room. The consumer can observe the visualization, and roughly figure out the budget they needed for his/her designated room type.

EDA deliverable: Minimum Nights vs. Room Type



This visualization shows how minimum nights varies with the room type. The analysis here is done to see if there are any weird information of minimum nights. You can see that the rooms are mostly within the 250-300 minimum night range. There is one apartment listing with 1000 minimum nights and private room with 5000 minimum nights which is ridiculous. I consider these two listings to be null. At this point, even listings with 200 minimum nights is not sensible for an average user, but I assume that these types of long night listings are for business men and workers who need a long stay.

This finding is crucial to developing our main tool (in calculating the total price including transportation price). Since the main tool will have users input their number of nights, we know that users are not going to find a listing with 100 minimum nights, which means that some of our listing data will not (never) be seen in most of times. If the user's goal is to work in London for example, they are unlikely to book a place with Airbnb. Most people would just directly book a place directly with the accommodation. +

Other Trivial EDA's that are performed

Distance(m) vs Uber price – to show that these two values are in positive correlation.

Neighborhood vs Accommodation – to show how many accommodations there are in a neighborhood

Accommodation vs Room type – to show how many accommodations there are for each of the rooms: Private Room, Share room, Apartment room

Data preparation for Tableau Public

With our current data, it is still not sufficient enough to export to Tableau yet. The existing raw data needs new parameters: Distance (from airport), Uber price, round trip price

Calculating Distance using Haversine Formula

Harversine formula is used specifically to calculate distance with given two coordinates (latitude, longitude). It is the shortest distance over the Earth's surface.

The formula is as follows:

Haversine formula:

$$a = \sin^2(\Delta\phi/2) + \cos\phi_1 \square \cos\phi_2 \square \sin^2(\Delta\lambda/2)$$

$$c = 2 \square \operatorname{atan2}(\square a, \square(1 \square a))$$

$$d = R \square c$$

where *ϕ is latitude, λ is longitude,* R *is earth's radius (mean radius = 6,371km);*
note that angles need to be in radians to pass to trig functions!

Developing our tool using analytics on Tableau Public

Aims and Objectives:

- Create a visualization that allows users to navigate through each accommodation to check prices (filters: price, neighborhood, round trip price)
- Create a map that contains all the accommodations. Each point on the map corresponds to the accommodation and upon interacting shows various information (filters: price, roundtrip price, neighborhood)

Overview of methodology

Tableau Public's user interface is very convenient to use. We first import the csv files that we created earlier into the software and now the data members will be available for access. We simply use those members to either insert into the rows tab or the columns tab on Tableau UI. This will determine the orientation of your visualization. Tableau has many kinds of visualizations to choose from so you can change from a bar graph to a pie graph for example. Now to allow users to interact with our visualizations (to become a tool), we add color (or legend) to allow users to distinguish stuff easier and filters.

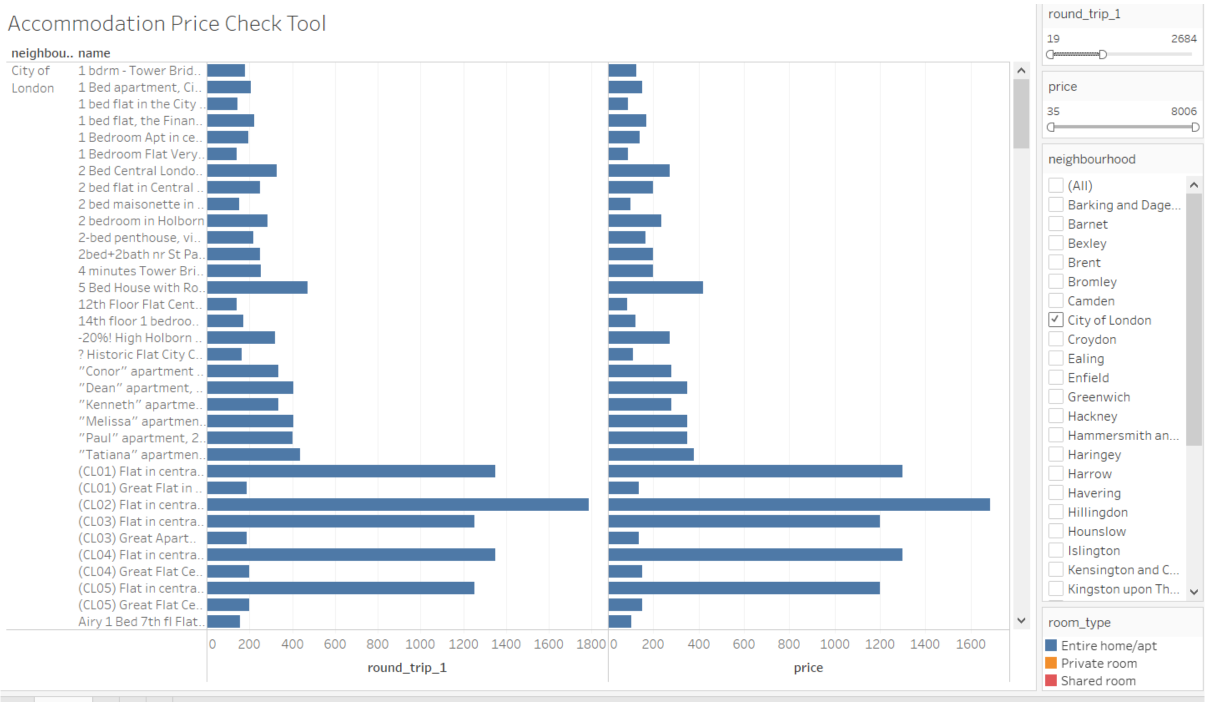
Price Check tool

Columns – round trip price, price

Rows- neighborhood, name (formed a hierarchy)

Legend/Colors – room type (3 colors = 3 room types)

Filters- round trip price (range), price (range), neighborhood



Accommodation Price Check Tool

round_trip_1
19 2684

price
35 8006

neighbourhood
□ (All)
□ Barking and Dage...
□ Barnet
□ Bexley
□ Brent
□ Bromley
□ Camden
☑ City of London
□ Croydon
□ Ealing
□ Enfield
□ Greenwich
□ Hackney
□ Hammersmith an...
□ Haringey
□ Harrow
□ Havering
□ Hillingdon
□ Hounslow
□ Islington
□ Kensington and C...
□ Kingston upon Th...

room_type
🟦 Entire home/apt
🟧 Private room
🟥 Shared room

User Interaction: Users can clearly compare prices between the round trip price and the price per night. The convenience lies in the presence of accommodation name list. Users can read names and follow along while observing prices. The legend here also clearly differentiates the different types of rooms.

Map UI Tool

Tableau provides users with map implementation and will automatically detect longitude and latitude data. Once it verifies the data, Tableau will generate a map according to the coordinates.

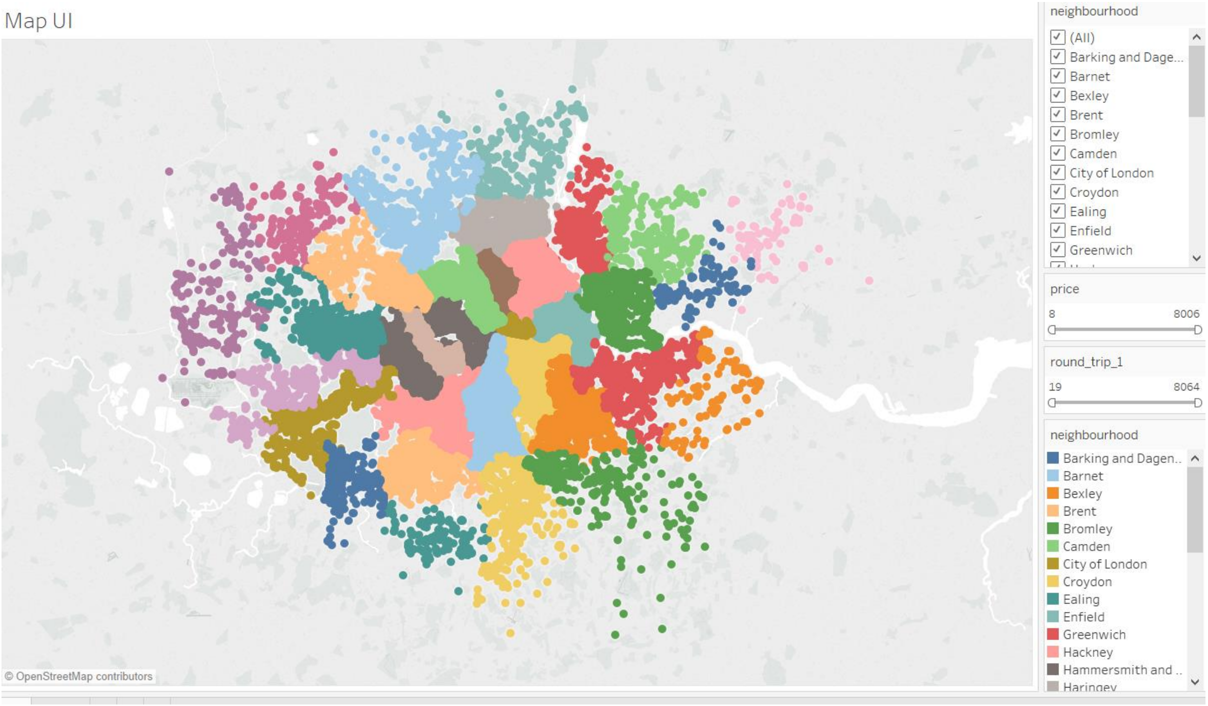
Columns – longitude

Rows- latitude

Legend/Colors – neighborhood (to easily distinguish each neighborhood)

Filters – round trip price (range), price (range), neighborhood

User Interaction - Users can adjust price ranges and their preferred neighborhood choice. Users can navigate over the points (accommodations) on the map to check valuable information such as the Uber Price and the distance in meters from Heathrow Airport.



neighbourhood

☑ (All)
☑ Barking and Dage...
☑ Barnet
☑ Bexley
☑ Brent
☑ Bromley
☑ Camden
☑ City of London
☑ Croydon
☑ Ealing
☑ Enfield
☑ Greenwich

price

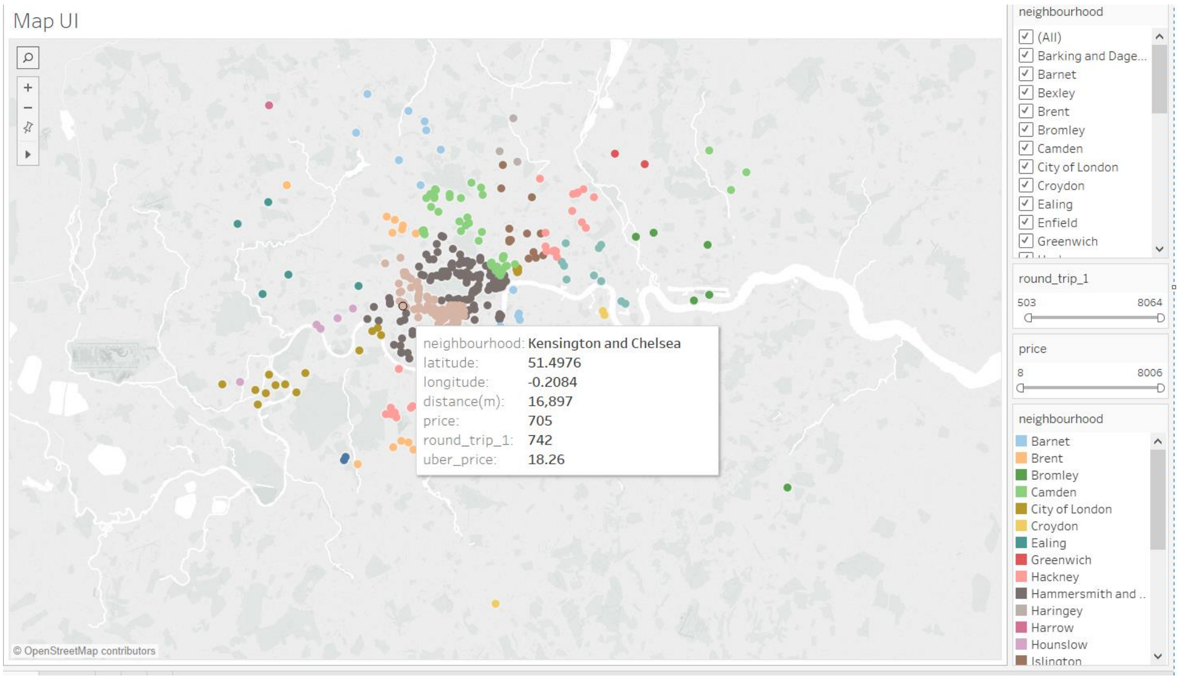
8 8006

round_trip_1

19 8064

neighbourhood

🟦	Barking and Dagen..
🟦	Barnet
🟧	Bexley
🟧	Brent
🟩	Bromley
🟩	Camden
🟨	City of London
🟨	Croydon
🟩	Ealing
🟩	Enfield
🟥	Greenwich
🟥	Hackney
■	Hammersmith and ..
■	Haringey



Map UI

neighbourhood

☑ (All)
☑ Barking and Dage...
☑ Barnet
☑ Bexley
☑ Brent
☑ Bromley
☑ Camden
☑ City of London
☑ Croydon
☑ Ealing
☑ Enfield
☑ Greenwich

round_trip_1

503 8064

price

8 8006

neighbourhood

■ Barnet
■ Brent
■ Bromley
■ Camden
■ City of London
■ Croydon
■ Ealing
■ Greenwich
■ Hackney
■ Hammersmith and ..
■ Haringey
■ Harrow
■ Hounslow
■ Islington

neighbourhood:	Kensington and Chelsea
latitude:	51.4976
longitude:	-0.2084
distance(m):	16,897
price:	705
round_trip_1:	742
uber_price:	18.26

Conclusion

This Airbnb tool project utilizes data wrangling, exploratory data analysis + storytelling, and statistics. It is designed to be an analytics tool for users to see different elements of an accommodation. What users will benefit most is how they can figure out their round trip budget and filter out accommodations accordingly. Users and UI are meant to be connected, which means users like seeing things visually, that's why a map UI is very effective.