Shahid Beheshti University

Department of Computer Science

Sample Report

Fundamentals of Data Science

Title

NYC Airbnb Open Data 2019; Pre-processing and Data Visualization

By

Student Name

Supervisor

Dr. Saeedreza Kheradpisheh

Month Year

Abstract

In this assignment the goal is to pre-process the *NYC Airbnb Open Data*[1] and visualize appropriate features and information. In a general view, first we pre-process the data by cleaning and dealing with the missing values, and next attempt to gain insight and extract knowledge from the data by visualizing it.

Keywords:

Data Science, Airbnb, Data Visualization, EDA, Pre-processing

Contents

Chapter 1

Introduction

As the first step it is essential to be clear and have a vivid vision of what we want from the data and have a prior sense of what might be hidden in the data; holding this view, we first pose some general questions:

- What can we learn about different hosts and areas?
- What can we learn from predictions? (ex: locations, prices, reviews, etc)
- Which hosts are the busiest and why?
- Is there any noticeable difference of traffic among different areas and what could be the reason for it?

These questions are mainly adopted from kaggle website [1]. Before we begin answering these questions, we should make sure that there is no null/Nan value, all the outliers are removed and etc.. This assurance is possible via imputing the null values, applying proper transformations on the features and so on. After those steps, to gain information about different hosts and areas, we begin plotting different graphs to answer this questions. Correlations between the given features, or predictions, can give us a bit of sense about the relations between each type of features. The reasons behind a specific host or area being busy can also be revealed by specific plots and methods which will be further discussed in the next chapter.

[1] Dataset address on kaggle website

Chapter 2

Methods

2.0.1 Pre-processing

The first step is to find out how many Nan/Null values the dataset has. The table 2.1 represents this information.

There are several ways to handle these missing values, such as:

- replacing them by the mean or median value of the field
- random value within the given range
- when the dataset is relatively large, ignoring the whole entry

in this case we fill the blanks with the most repeated value of that field (mode). Also for the sake of simplicity and respecting the privacy of the hosts, first we drop *name*, *id*, *host_name* columns. In order to deal with the outliers and invalid data, we drop those entries that have an absolute z-score of more than three. Figure 2.1 and figure 2.2 depict the boxplot of each numerical column of the dataset, before and after applying the less-than-three-z-score rule.

As we can see, the second figure demonstrates smaller values and less outliers after removing previous ones.

Normality Tests and Transformations

As a preprocessing step, it is better to transform values of a field that don't have a normal distribution, if possible. First we introduce some measures of normality and then some transformations to make non-normal distributions normal.

NYC Airbnb Listings	
Column Name	Missing Values
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

Table 2.1: Number of missing values, Nan/Null, in the NYC Airbnb dataset

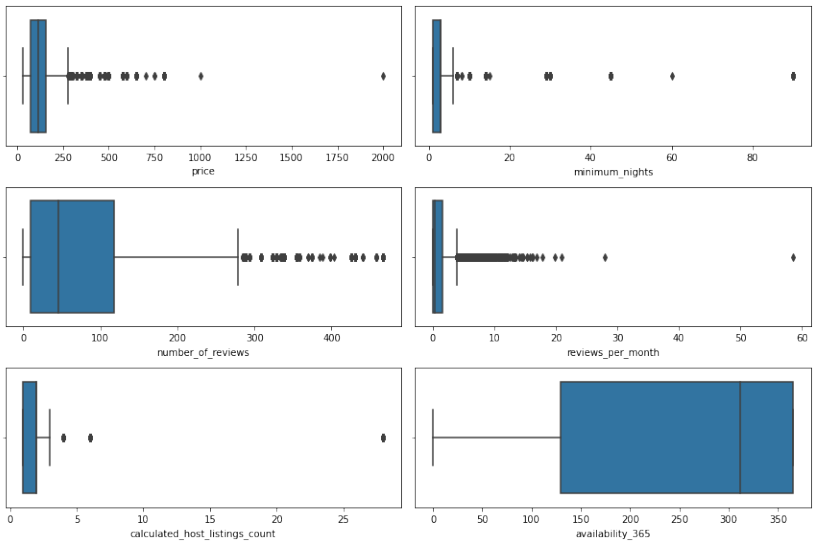

Figure 2.1: Features' boxplots before removing outliers

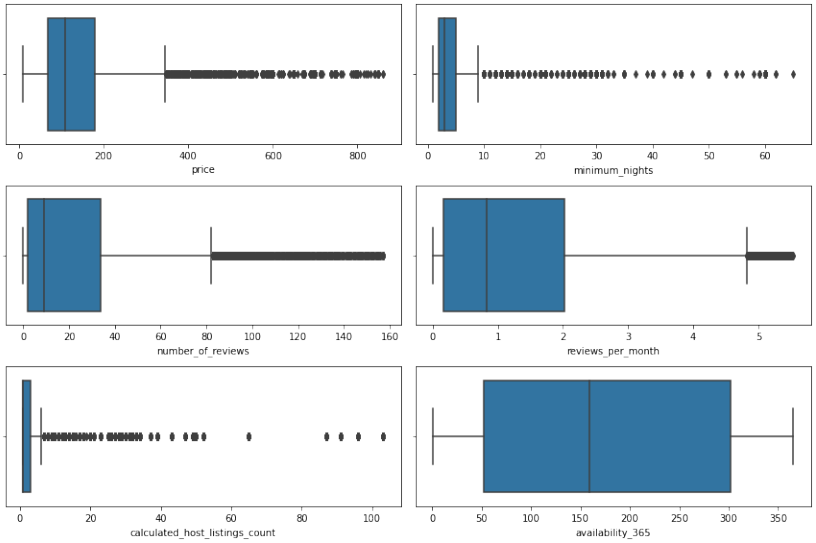

Figure 2.2: Features' boxplots after removing outliers

Skewness

In statistics, skewness is a measure of the asymmetry of the probability distribution of a random variable about its mean. In other words, skewness tells you the amount and direction of skew (departure from horizontal symmetry). The skewness value can be positive or negative, or even undefined. If skewness is 0, the data are perfectly symmetrical, although it is quite unlikely for real-world data. As a general rule of thumb: If skewness is less than -1 or greater than 1, the distribution is highly skewed. If skewness is between -1 and -0.5 or between 0.5 and 1, the distribution is moderately skewed. If skewness is between -0.5 and 0.5, the distribution is approximately symmetric. This statistics is defined as:

$$S = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3}{(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2)^{3/2}} \tag{2.1}$$

Here, $\bar{x}$ is the sample mean.

Kurtosis

Kurtosis reflects the height and sharpness of the central peak, relative to that of a standard bell curve, and is calculated as:

$$K = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^4}{(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2)^2} - 3 \tag{2.2}$$

Here, $\bar{x}$ is the sample mean. The "minus 3" at the end of this formula is often explained as a correction to make the kurtosis of the normal distribution equal to zero, as the

kurtosis is 3 for a normal distribution.[1]

Normal Test

The following Normal Test is a package-defined test based on D'Agostino and Pearnson's test, developed by SciPy, and is formulated as:

$$S^2 + K^2 \tag{2.3}$$

where s is the z-score returned by Skew test and k is the z-score returned by Kurtosis test.

Transformations

There are several methods that transforms distributions, i.e. skewed dist., to a normal one which we will further discuss and apply to our data, such as:

- Yeo-Johnson

$$x_i^{(\lambda)} = \begin{cases} [(x_i + 1)^\lambda - 1]/\lambda & \text{if } \lambda \neq 0, x_i \geq 0, \\ \ln(x_i + 1) & \text{if } \lambda = 0, x_i \geq 0 \\ -[(-x_i + 1)^{2-\lambda} - 1]/(2 - \lambda) & \text{if } \lambda \neq 2, x_i < 0, \\ -\ln(-x_i + 1) & \text{if } \lambda = 2, x_i < \end{cases} \tag{2.4}$$

- Box-Cox

$$x_i^{(\lambda)} = \begin{cases} \dfrac{x_i^\lambda - 1}{\lambda} & \text{if } \lambda \neq 0, \\ \ln(x_i) & \text{if } \lambda = 0, \end{cases} \tag{2.5}$$

- Logarithm

Now we show the results of the best transformation in terms of reduction in score of each test, which is the Box-Cox transformation. Figure 2.3 demonstrates the distribution of numerical columns before applying the Box-Cox transformation and the table 2.2 represents the statistical test results. Figure 2.4 illustrates each feature's

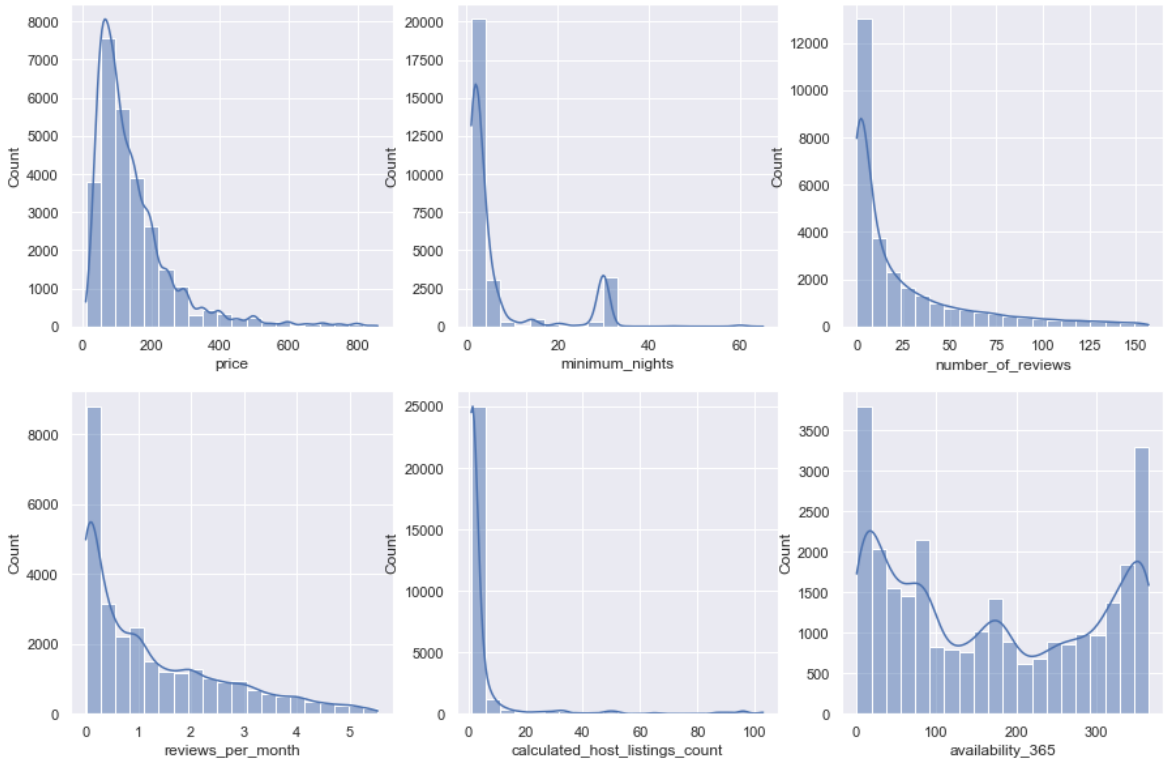

Figure 2.3: Features' boxplots before applying transformations

distribution after applying the transformation, also we can see that the scores have gotten better, in terms of normality, in table 2.3.

Test Name	NormalTest		SkewTest		KurtosisTest	
Feature Name	Score	p-value	Score	p-value	Score	p-value
price	15438.20	0.0	104.37	0.00	67.40	0.0
minimum nights	69228.99	0.0	230.69	0.00	126.52	0.0
number of reviews	20388.91	0.0	118.83	0.00	79.16	0.0
reviews per month	23466.99	0.0	117.38	0.00	98.42	0.0
calculated host listings count	35716.92	0.0	161.05	0.00	98.88	0.0
availability 365	204487.71	0.0	9.45	3.1e-21	452.10	0.0

Table 2.2: Statistical tests results' before applying transformations

[1]source

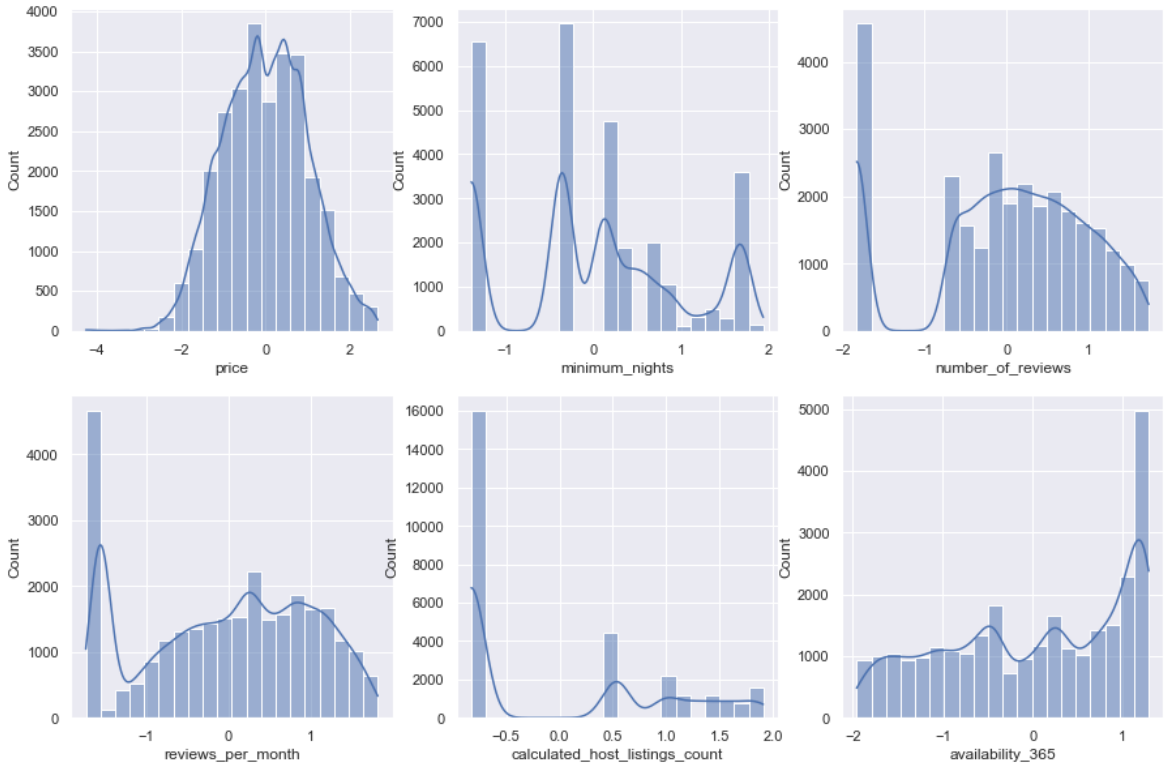

Figure 2.4: Features' boxplots after applying transformations

Test Name	NormalTest		SkewTest		KurtosisTest	
Feature Name	Score	p-value	Score	p-value	Score	p-value
price	183.41	1.4e-40	0.71	0.47	-13.52	1.12e-41
minimum nights	5339.61	0	13.52	1.10-41	-71.80	0
number of reviews	1615.13	0	-32.13	1.5e-226	-24.13	1.01e-128
reviews per month	7531.92	0	-11.46	1.9e-30	-86.02	0
calculated host listings count	31765.7	0	40.39	0	-173.59	0
availability 365	28510.3	0	-22.71	3.4e-114	-167.31	0

Table 2.3: Statistical tests results' after applying transformations

Test Name	NormalTest		SkewTest		KurtosisTest	
Feature Name	Score	p-value	Score	p-value	Score	p-value
Random normal dist.	0.81	0.66	0.70	0.48	0.56	0.57

Table 2.4: Statistical tests results' of the random normal distribution mentioned in figure 2.5

Overall, although scores are not in the aforementioned range for the normal distributions, but they have been reduced substantially. Just to get a better sense of normal distribution and it's scores, figure 2.5 shows a normal distribution and table 2.4 represents it's scores.

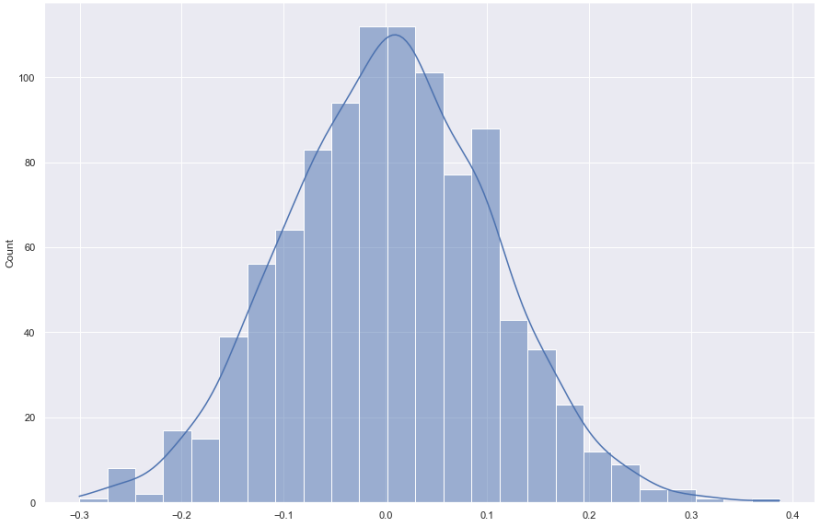

Figure 2.5: A random normal distribution graph

2.0.2 Visualization

Geographical Visualization

Now it's time to deal with the geographical data, like latitude and longitude and neighbourhood details. We can derive new a column, 'density', from the given geographical coordinates to illustrate a heatmap-like graph of the density of accommodations. This density column is a kernel-density estimate of the given coordinates using gaussian kernels and is calculated by scipy's 'gaussian_kde' class. Also the size of each circle is relative to that home's price. Figure 2.6 shows a sample of 10000 houses on the Airbnb NYC listings.

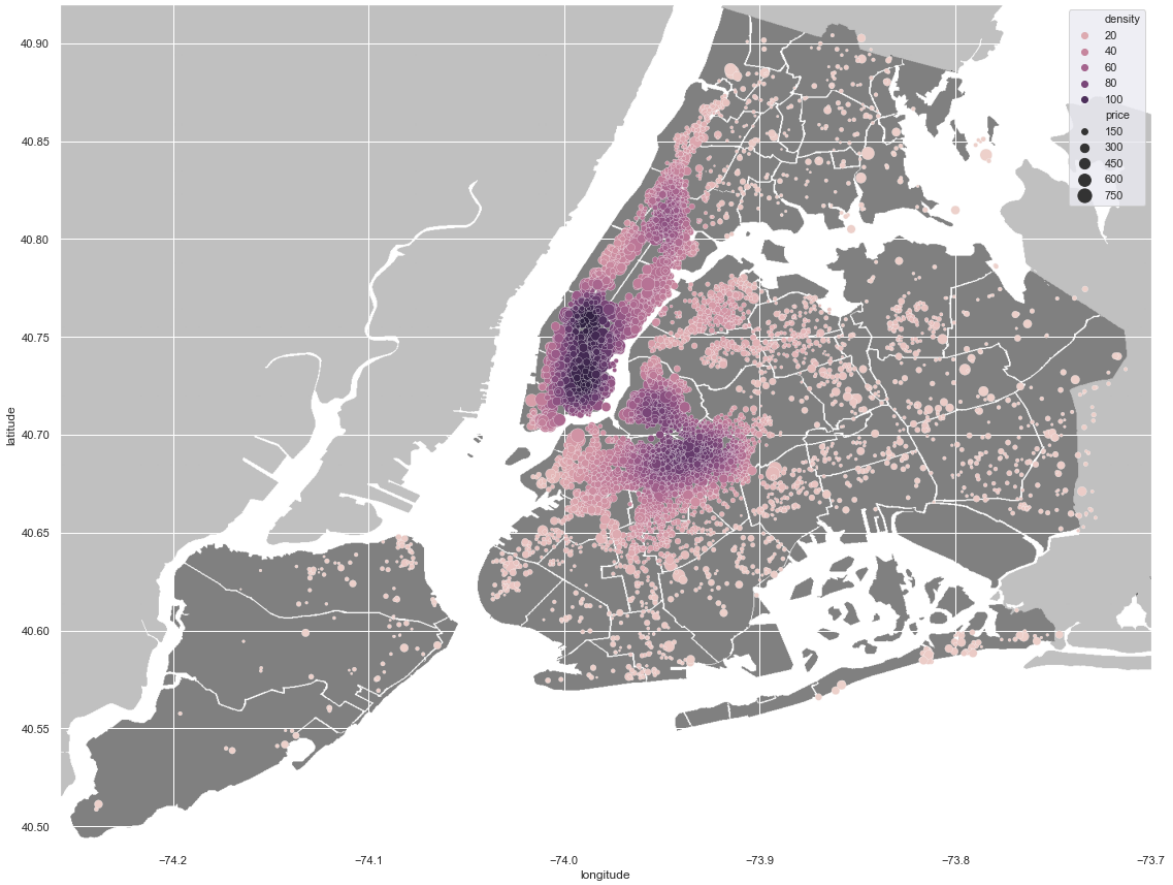

Figure 2.6: Listing of Airbnb houses in NYC, the larger and darker dots correspond to the price and density of a single listing respectively

For investigating the reasons of this density in certain areas of the NYC, it is reasonable to collect the recreational and entertainment places location and plot them on the map, which has been done and clustered in the figure 2.7. This figure mainly contains locations of shops, restaurants, and museums.

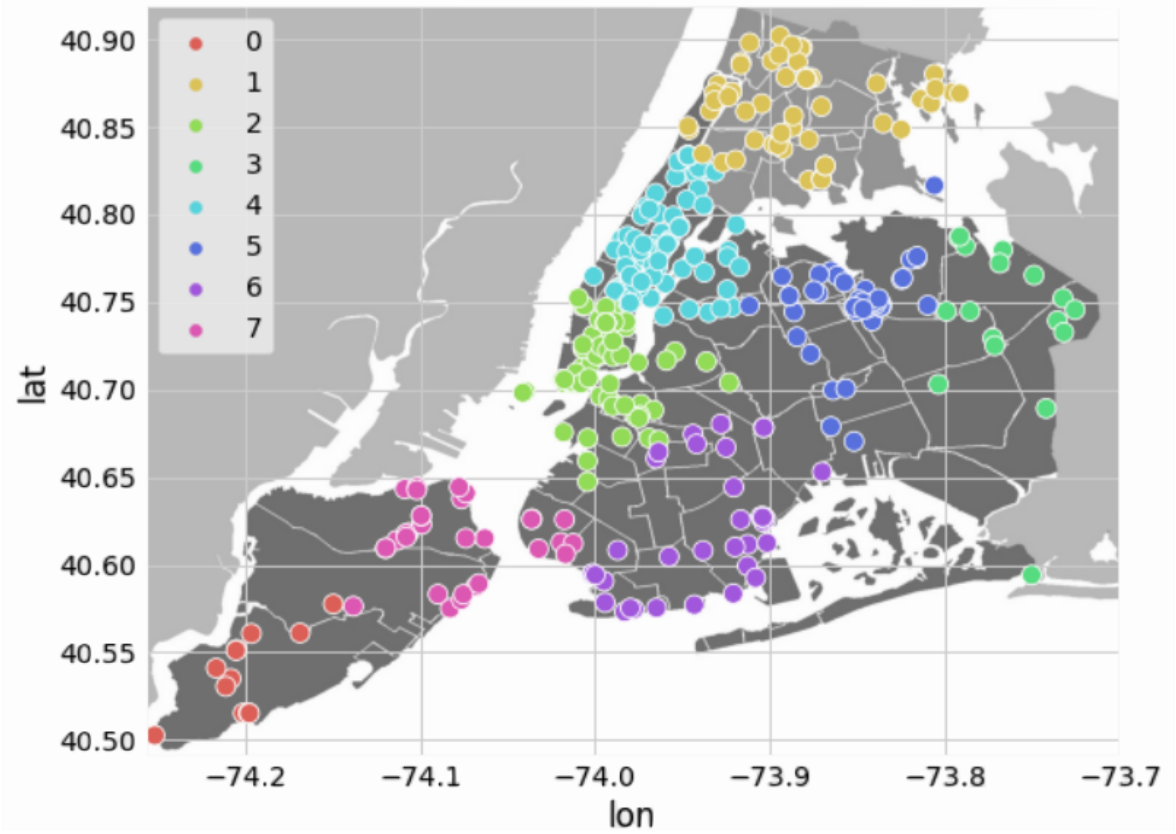

Figure 2.7: Location of museums, restaurants, and shops in NYC

It is also desirable to have a pie-chart depicting the percentage of each neighbourhood's listings in the overall list, as figure 2.8 suggests.

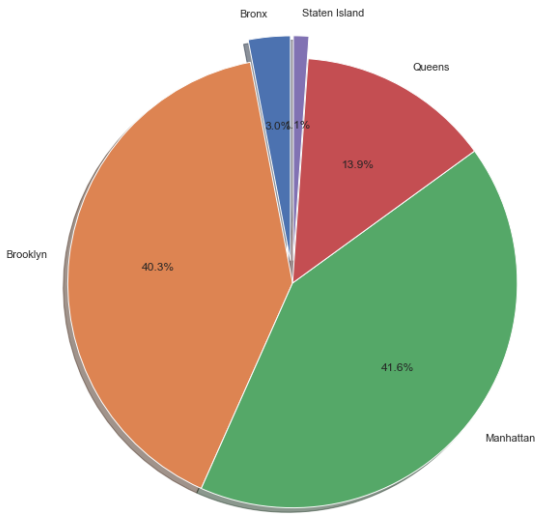

Figure 2.8: Distribution of NYC Airbnb accommodations among 5 major neighbourhood groups

Next we group the dataset by ‘neighbourhood‘ and plot each individual neighbourhood’s price, as figure 2.9 demonstrates, circle’s radius corresponds to the average price of an accommodation in that area.

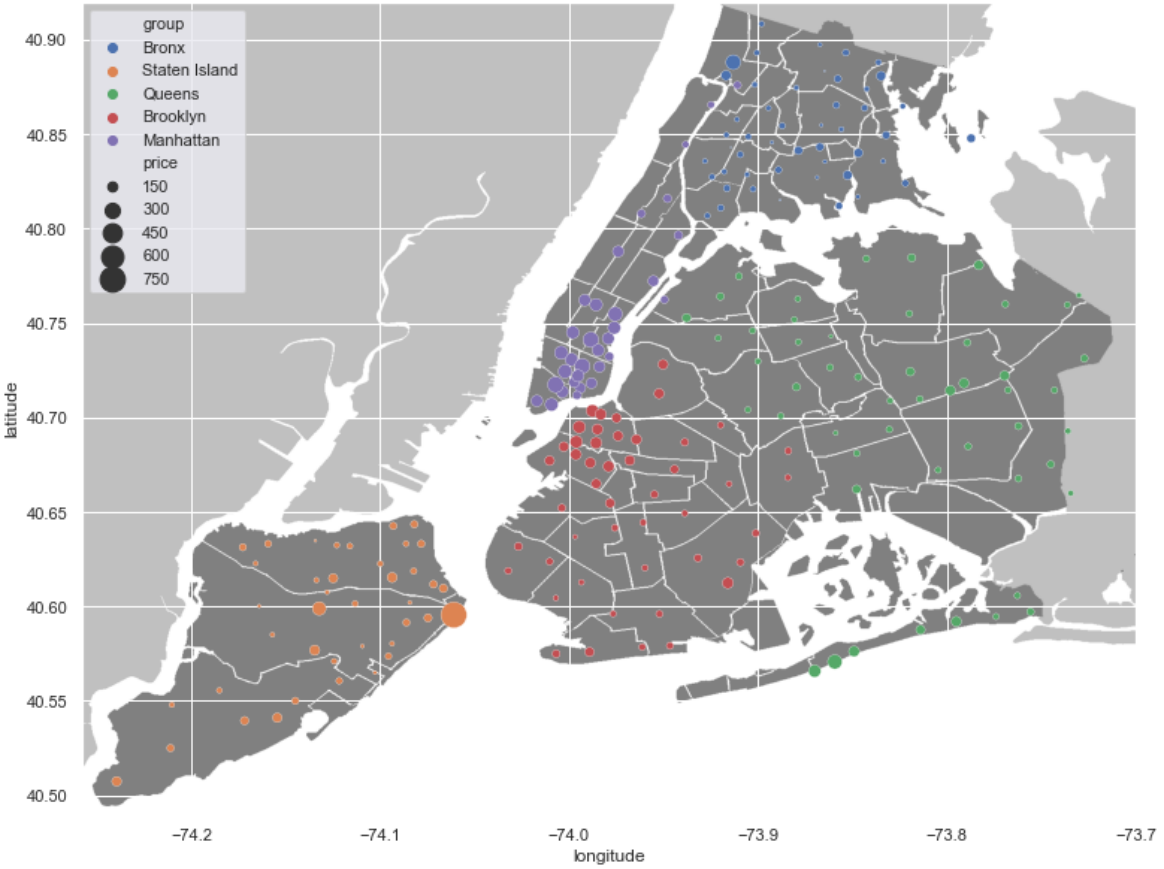

Figure 2.9: Every single neighbourhood’s average price in NYC

Also it is possible to find those hosts that are the busiest by applying a threshold on the entries that have listed homes more than the others. First we need to group the dataset by ‘host_id‘ to have summarized information about each one. After grouping, we sort the hosts by the ‘calculated_host_listings_count‘ to identify the busiest hosts via applying a threshold selecting only those who have listed more than the average, as figure 2.10 illustrates.

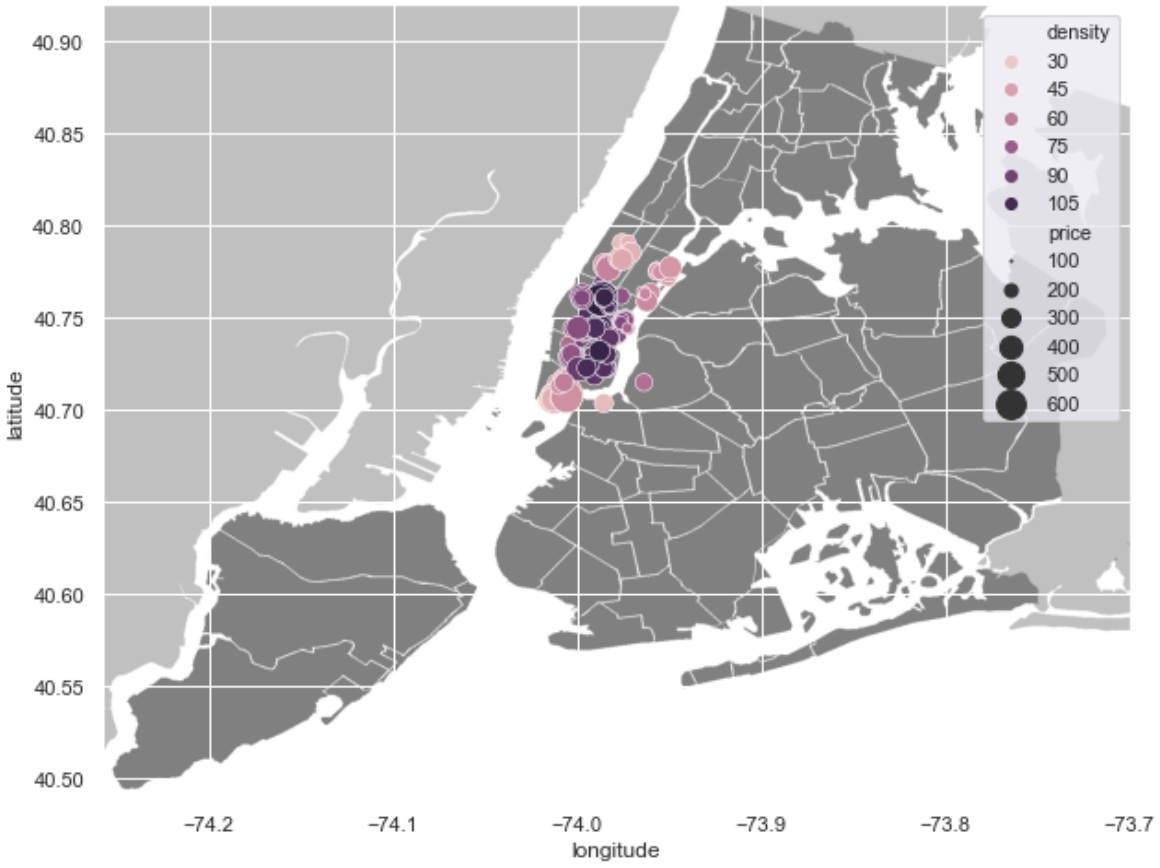

Figure 2.10: Every single neighbourhood center in NYC

In terms of different types of rooms, by grouping the dataset, we can see that 'price', 'minimum_nights', 'number_of_reviews', and 'calculated_host_listings_count' columns, table 2.5, are different within each group, so we plot those, figure 2.11.

Type of Room	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_listings_count	availability_365	density
Entire home/apt	201.21	10.32	31.30	1.45	15.90	170.96	57.97
Private room	86.41	5.96	33.72	1.61	4.22	177.42	47.93
Shared room	62.75	6.97	19.20	1.32	5.56	217.65	47.18

Table 2.5: Average of features based on the accommodation type

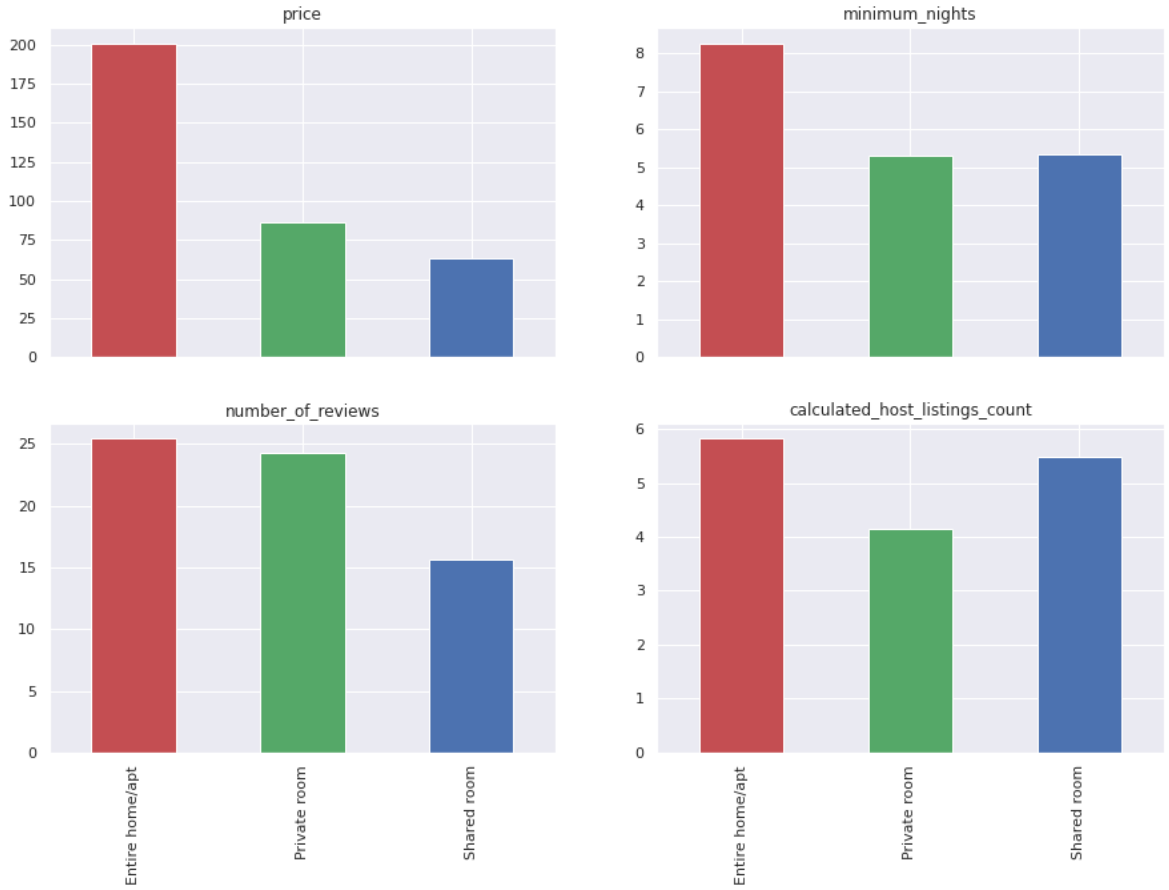

Figure 2.11: Features categorized by property types

Also we can plot different types of properties located within major neighbourhood groups. Here we plot ‘calculated_host_listings_count‘, ‘price‘, and ‘density‘ columns by ‘sns.barplot‘. By specifying the column name as ‘hue‘ parameter, you can get more detailed bars, as we’ll see this for the ‘room_type‘ column, figure 2.12.

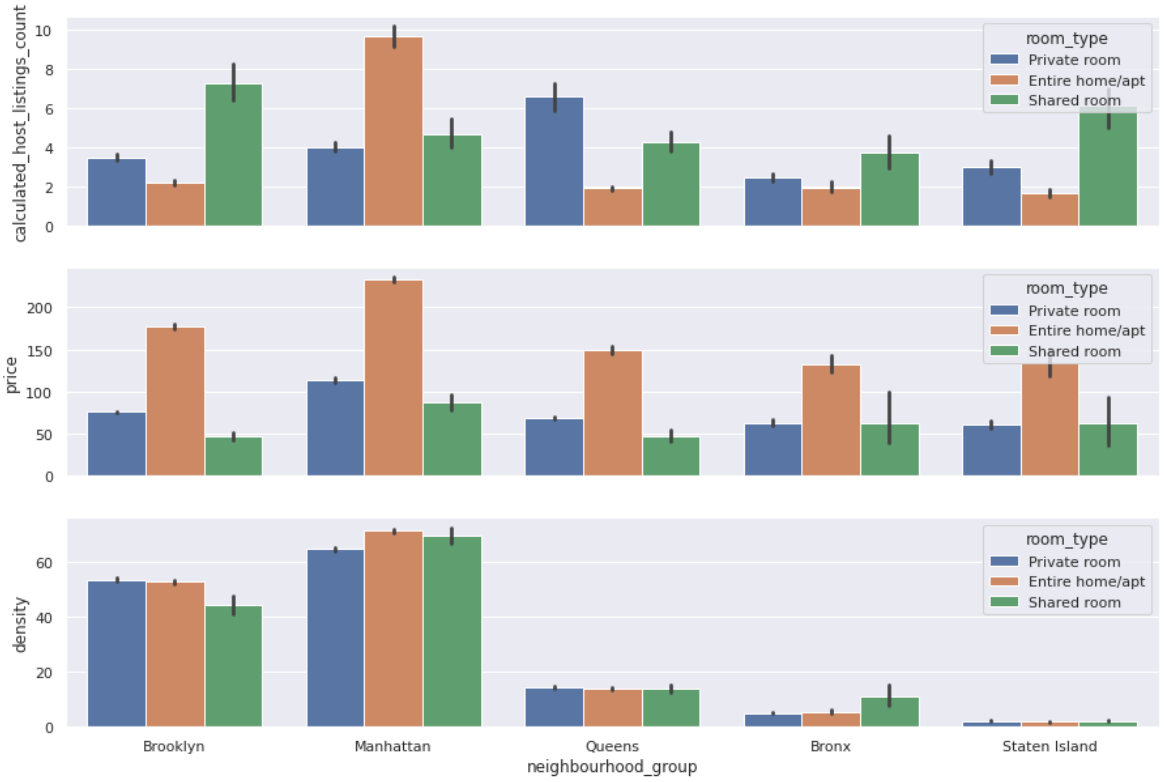

Figure 2.12: Density and types of accommodations within major neighbourhood groups in NYC

2.0.3 Prediction

There are features in this dataset that are likely to be correlated, ‘reviews_per_month‘ and ‘number_of_reviews‘ are an obvious example. Hence by plotting the correlation matrix, using a threshold, we can gain a nice perspective of how these variables will behave respectively. For instance, it’s apparent that as the density of a place grows, the prices will consequently rise up, 2.13.

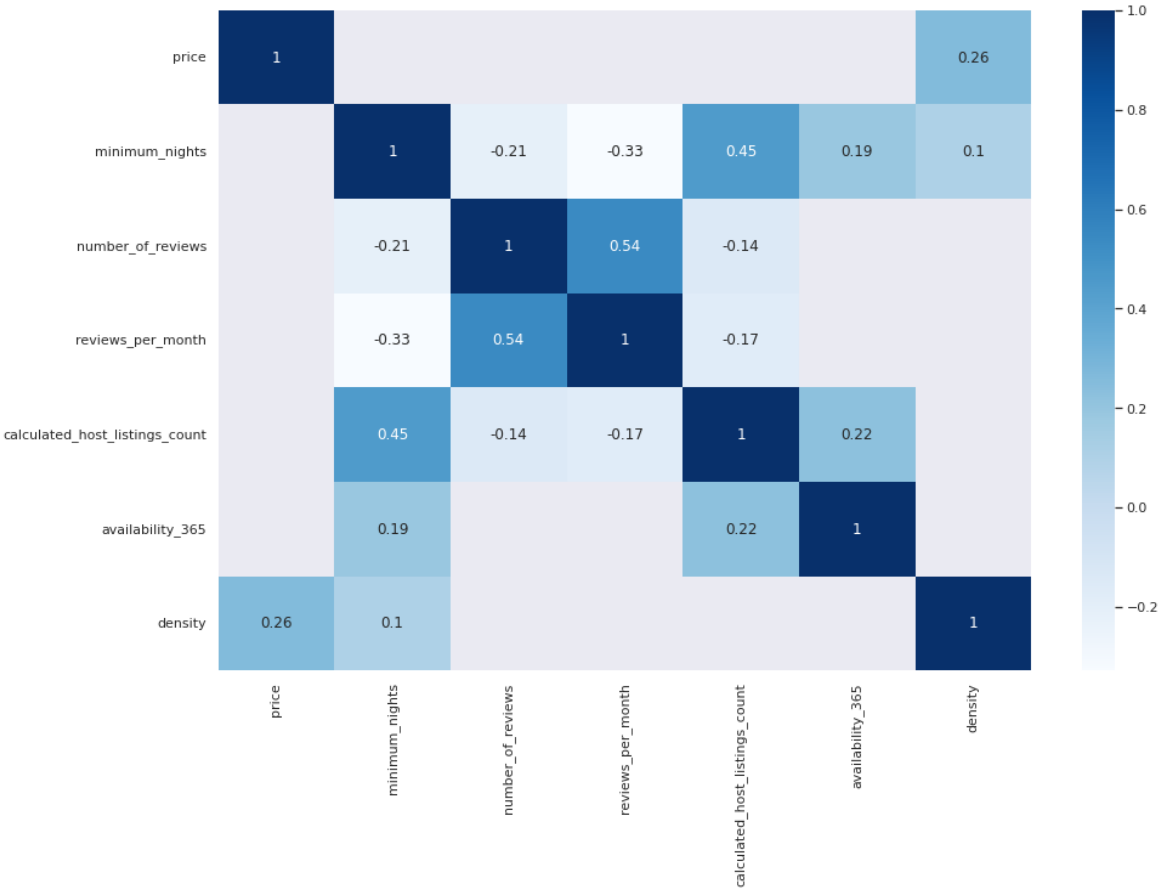

Figure 2.13: Every single neighbourhood center in NYC

Chapter 3

Conclusion

After performing EDA on the Airbnb NYC listings dataset, we almost addressed the questioned proposed in the introduction section and now can state few key points about Airbnb accommodations, mainly:

- There are certain areas that gain interest of travelers and tourists, figure 2.6.
- The density of these accommodations are positively correlated with the density of shops, restaurant and other facilities, figure 2.7.
- Brooklyn and Manhattan are the two most attractive neighbourhoods for the guests, figure 2.8.
- Average rent price of neighbourhoods are related with the density of each one, figure 2.9.
- Busiest hosts' listings are located in Manhattan, figure 2.10
- Minimum nights and price of an entire home/apartment are significantly higher than the other two types, figure 2.11.
- Entire home listings are more popular, and expensive, in Manhattan than the other four neighbourhood groups, figure 2.12.
- Prices of Entire homes are the highest among other type in all neighbourhoods, and shared rooms have an approximately equal price with the private rooms in all major hoods, figure 2.12.
- Not surprisingly, those files that are located in the more dense areas have higher prices, and vice versa; Whereas, surprisingly, those houses that are provided by a "busy host" are more likely to be available and expensive, figure 2.13.

- The more items a host provides, the more nights are set as minimum, figure 2.13.
- As the number of minimum nights decreases, the review per months of that house increases, figure 2.13.

Chapter 4

References

[1] Inside Airbnb, The dataset describes the listing activity and metrics in NYC, NY for 2019. http://insideairbnb.com/new-york-city/