Initial Analysis of U.S. Airbnb Open Data

Peker Milas

November 29, 2020

Contents

1 Summary of the Project **2**

2 Description of the Data **2**

3 Initial Analysis Plan **2**

4 Data Cleaning and Features **3**

5 Current Findings **5**

6 Next Steps and Hypothesis **7**

6.1 Statistical Test on Effect of City Feature 8

7 About the Quality of the Data **9**

1 Summary of the Project

Airbnb can be sen as a game changer in the traditional hospitality industry thanks to its impressive flexibility in a variety of ways. Starting from the price ranges to the diverse location options it offers, Airbnb is a go-to service which many travellers decide to use as their primary means of accommodation. With the exception of COVID era, it is certainly a rapidly growing hosting service, so I believe it is worth to get some insights about how it works. Therefore I decided to analyze the data provided on **Kaggle** Section 2, with the aim of estimating Airbnb prices by using the remaining features of the data in case they correlated enough to come up with a model.

2 Description of the Data

The data[1] provided on Kaggle is a CSV file with the name AB_US_2020.csv. It has columns describing features namely listing id, description(called **name**), host id, host name, neighborhood, ZIP code, latitude, longitude, room type, price, minimum number of nights, number of reviews, last review date, reviews per month, host listings count, availability in days, and city. Among these features, name, host name, neighborhood, room type, and city are categorical while the rest are numerical features.

The data consist of 226029 unique entries with a total of 14 features. Only the features name, last review date, and reviews per month have missing values. Particularly the latter two have 48602 missing values which need to be processed accordingly during the data clean-up.

3 Initial Analysis Plan

My initial analysis plan consisted on following steps;

1. Load the data to a data-frame by initially ignoring ZIP code and neighborhood features to reduce the complexity relying on the idea that they would be already coupled into other features such as latitude, longitude, and city.
2. Identify and filter out categorical features in the data for further simplification and to see if there is obvious correlations among numerical features.
3. Identify and process missing data within numerical features.
4. Visualize numerical data for checking the existence of outliers.
5. Identify and filter out outliers.
6. Visualize the cleaned-up numerical data for correlations and obvious relations.

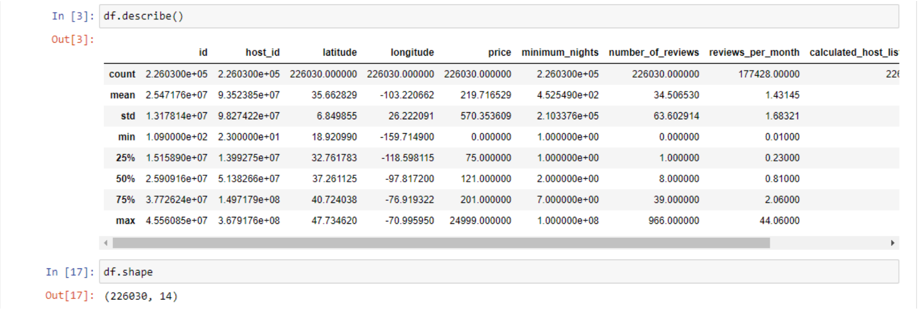
[1] `https://www.kaggle.com/kritikseth/us-airbnb-open-data`

7. If there is enough correlation among price and the remaining features, proceed to the modeling the price feature using regression techniques.

8. Otherwise utilize redo the analysis by utilizing categorical features too.

4 Data Cleaning and Features

Following the analysis plan in Section 3, I decided to discard missing data. Original data and the data after the removal of missing values are shown in Figures 1, and 2

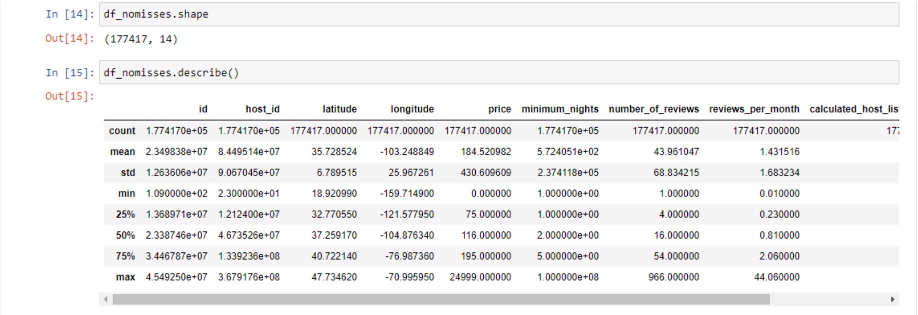


```
In [3]: df.describe()
Out[3]:
```

	id	host_id	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_lis
count	2.260300e+05	2.260300e+05	226030.000000	226030.000000	226030.000000	2.260300e+05	226030.000000	177428.00000	22
mean	2.547176e+07	9.352385e+07	35.662829	-103.220662	219.716529	4.525490e+02	34.506530	1.43145	
std	1.317814e+07	9.827422e+07	6.849855	26.222091	570.353609	2.103376e+05	63.602914	1.68321	
min	1.090000e+02	2.300000e+01	18.920990	-159.714900	0.000000	1.000000e+00	0.000000	0.01000	
25%	1.515890e+07	1.399275e+07	32.761783	-118.598115	75.000000	1.000000e+00	1.000000	0.23000	
50%	2.590916e+07	5.138266e+07	37.261125	-97.817200	121.000000	2.000000e+00	8.000000	0.81000	
75%	3.772624e+07	1.497179e+08	40.724038	-76.919322	201.000000	7.000000e+00	39.000000	2.06000	
max	4.556085e+07	3.679176e+08	47.734620	-70.995950	24999.000000	1.000000e+08	966.000000	44.06000	

```
In [17]: df.shape
Out[17]: (226030, 14)
```

Figure 1: Initial data and its brief description.



```
In [14]: df_nomisses.shape
Out[14]: (177417, 14)

In [15]: df_nomisses.describe()
Out[15]:
```

	id	host_id	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month	calculated_host_lis
count	1.774170e+05	1.774170e+05	177417.000000	177417.000000	177417.000000	1.774170e+05	177417.000000	177417.000000	17
mean	2.349838e+07	8.449514e+07	35.728524	-103.248849	184.520982	5.724051e+02	43.961047	1.431516	
std	1.263606e+07	9.067045e+07	6.789515	25.967261	430.609609	2.374118e+05	68.834215	1.683234	
min	1.090000e+02	2.300000e+01	18.920990	-159.714900	0.000000	1.000000e+00	1.000000	0.010000	
25%	1.368971e+07	1.212400e+07	32.770550	-121.577950	75.000000	1.000000e+00	4.000000	0.230000	
50%	2.338746e+07	4.673526e+07	37.259170	-104.876340	116.000000	2.000000e+00	16.000000	0.810000	
75%	3.446787e+07	1.339236e+08	40.722140	-76.987360	195.000000	5.000000e+00	54.000000	2.060000	
max	4.549250e+07	3.679176e+08	47.734620	-70.995950	24999.000000	1.000000e+08	966.000000	44.060000	

Figure 2: The data after removal of missing values its brief description.

Visualization of the data after the removal of missing values is done by histograms as shown in Figure 3. It was obvious that outliers exist at least in the features number of reviews, reviews per month, minimum available nights, and host listings counts.

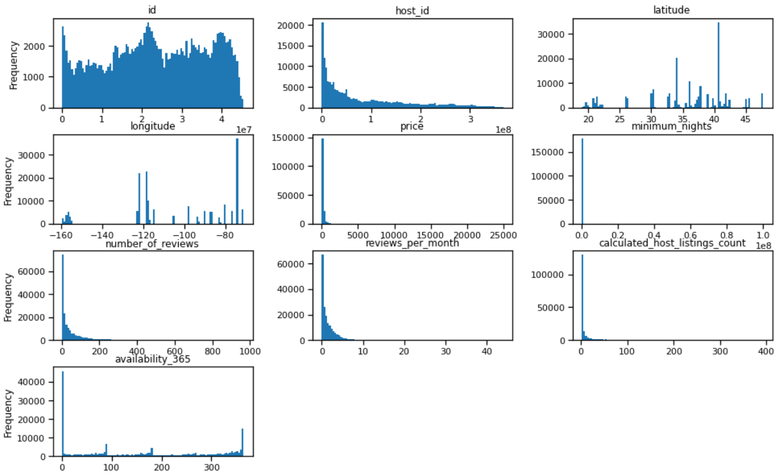


Figure 3: Histograms of data after removal of missing values.

Outliers removed by applying the following constraints;

- Price feature will be accepted as long as it is between first and third quartile.
- Number of reviews feature will be accepted as long as it is between 0 and 200 (both exclusive).
- Host listings count feature will be accepted as long as it is smaller than 25.
- Minimum nights feature will be accepted as long as it is smaller than 10.
- Reviews per month feature will be accepted as long as it is smaller than zero.
- Number of available nights feature will be accepted as long as it is between 0 and 365 (both exclusive).

With these constraints applied, data after removal of outliers looked quite reasonable as shown in Figure 4.

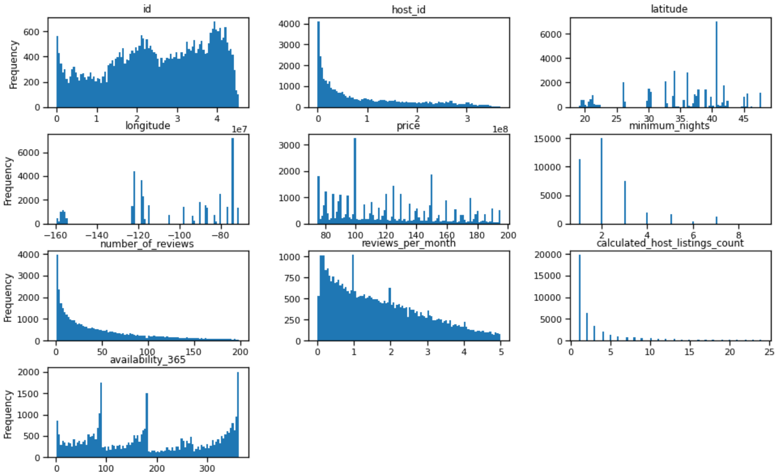


Figure 4: Histograms of data after removal of outliers.

5 Current Findings

Pair-plots of features on the cleaned-up data showed not obvious correlations between price and the remaining features as shown in Figure 5.

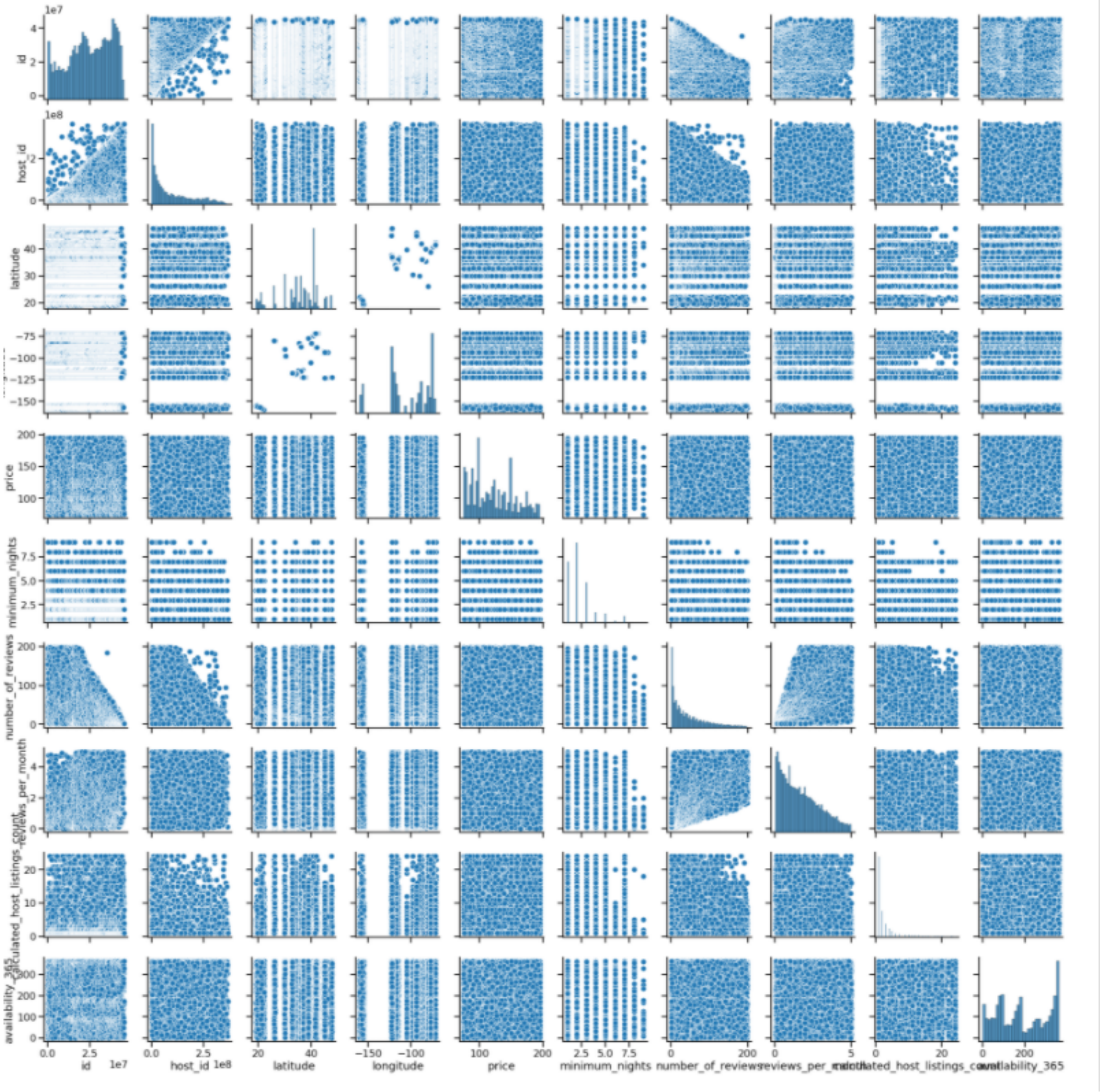


Figure 5: Pair-plots of features.

On the other hand small correlations observer between price and other features within the range of ±0.11 to ±0.058 with the except of host id and id features (≤ ±0.016). Calculated Pearson correlations are shown in Figure 6.

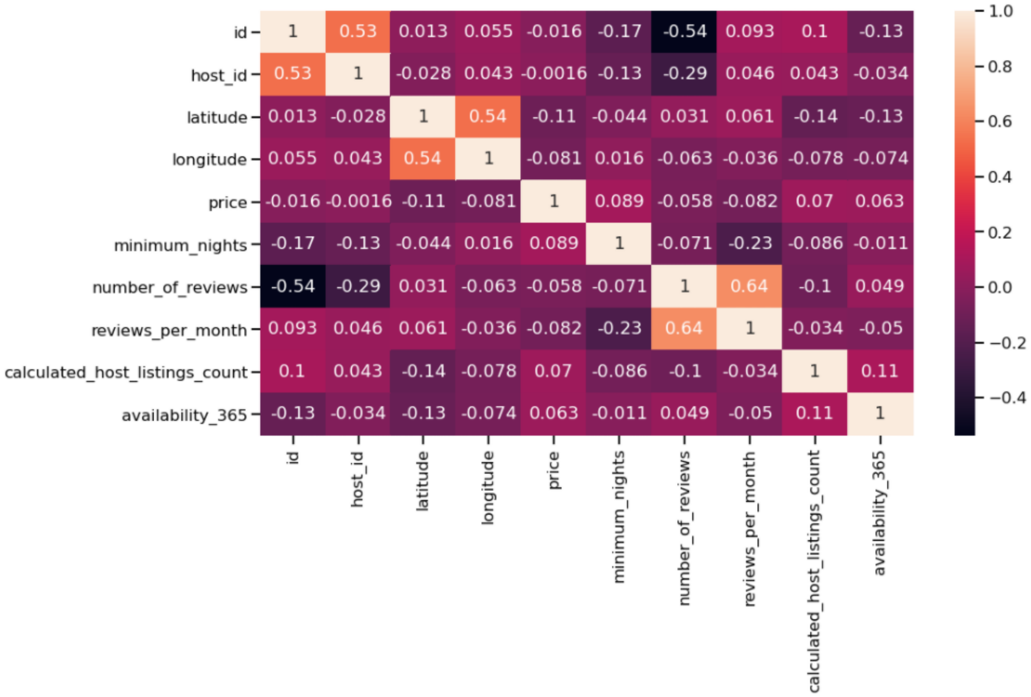


Figure 6: Pearson correlation-plots of features.

6 Next Steps and Hypothesis

Observed weak correlations between price and remaining features may indicate the requirement of categorical features within the analysis. Based on this, in the next steps of my analysis I would repeat the previously described procedure on city basis. If I observe a clear correlation after city based grouping of data, the next logical step will be application of linear regression on the data. Putting all together, my hypothesis for upcoming analysis steps will be as following;

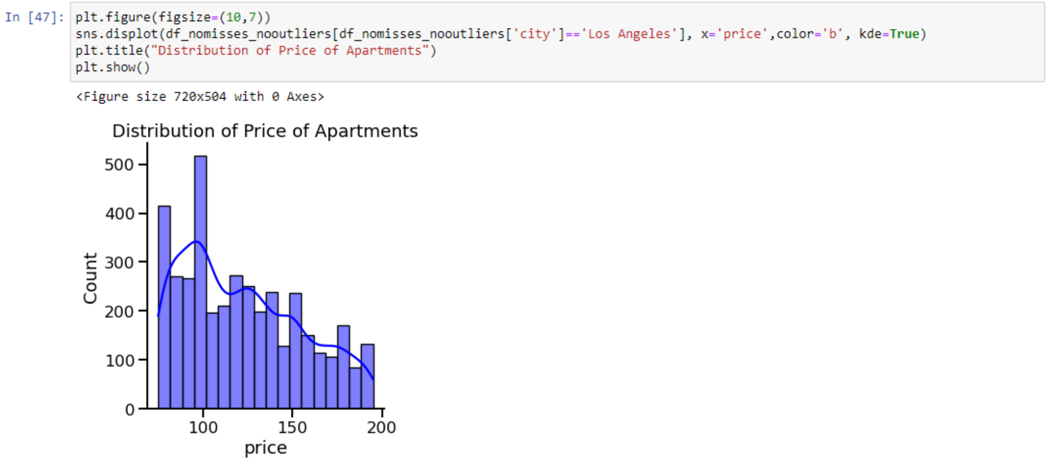
- City based group data will show stronger correlations with the remaining features.
- Requirement of including another categorical feature such as ZIP code or neighborhood will be obvious if the distribution price feature is not normal.
- To model the price correctly, I need consider city sizes too.

6.1 Statistical Test on Effect of City Feature

To provide an example for testing grouped data by city, I would be checking the normality of the distribution of prices within a city. My hypothesis and what they will refer to in actual data will be as following;

- $\mathbf{H_0}$ (Null Hypothesis): Price distribution is normal within a city of choice indicating that only 1 categorical feature namely 'city' is enough to describe the price.
- $\mathbf{H_1}$ (Alternate Hypothesis): Price distribution is not normal within a city of choice indicating that only I may need more than 1 categorical feature namely 'city' to describe the price.

To test the normality of price distribution in Los Angeles shown in Figure 7, I use Shapiro test[2].



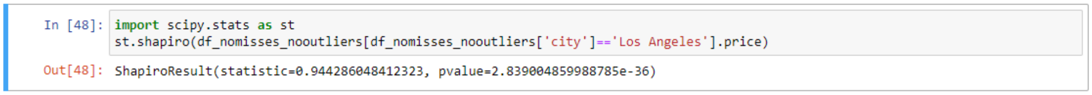
```
In [47]: plt.figure(figsize=(10,7))
         sns.displot(df_nomisses_nooutliers[df_nomisses_nooutliers['city']=='Los Angeles'], x='price',color='b', kde=True)
         plt.title("Distribution of Price of Apartments")
         plt.show()

<Figure size 720x504 with 0 Axes>
```

Figure 7: Histogram of prices in Los Angeles.

Using Scipy' s stats library and assuming a 5% alpha value, I found a p-value of 2.839004859988785e-36 as shown in Figure 8. Based on the test result, I can safely reject the Null Hypothesis. In other words, I will most likely need to utilize other categorical features to describe prices in the data.

[2] https://en.wikipedia.org/wiki/Shapiro%E2%80%93Wilk_test



```
In [48]: import scipy.stats as st
         st.shapiro(df_nomisses_nooutliers[df_nomisses_nooutliers['city']=='Los Angeles'].price)

Out[48]: ShapiroResult(statistic=0.944286048412323, pvalue=2.839004859988785e-36)
```

Figure 8: Test of normality of prices in Los Angeles.

7 About the Quality of the Data

This is a real data set, so I was expecting to have more missing values if not more outliers. On the other hand, I found out that around 90% of the data were usable. This is actually the only comment I can make about the data before I include categorical features into my analysis. on the over all quality of the data.