Data1030_Project_Proposal

Yuyang Li

Airbnb in New York

1. Project Task

The goal of this project is to build machine learning models to predict the prices of Airbnb in New York, given the datasets from Kaggle. See https://www.kaggle.com/dgomonov/new-york-city-airbnb-open-data/downloads/new-york-city-airbnb-open-data.zip/3#AB_NYC_2019.csv. The target variable is the prices of Airbnb in New York and the problem is regression. It is pretty interesting because nowadays more people use airbnb to expand on traveling possibilities. Staying in airbnb gives people more personalized experiences.

Reference: https://github.com/mikelyy/Airbnb-in-New-York

2. Related Work

From Kaggle Kernels, I have gone through some of projects completed by other people. In Dgomonov's "Data Exploration on NYC Airbnb", he uses the data for security, business decisions and guiding marketing initiatives, etc. Features such as id, host_name and last_review are dropped since these features are not significant for future predictions. I get some ideas of how to implement exploratory data analysis and feature engineering for the dataset. In Andrew W's "Smart Pricing with XGB, RFR + Interpretations", he uses the data to build and train a smart pricing model. The target variable is price. The price is initially transformed through log-transformation. I learn some feasible regression models.

3. Dataset and Metric

The dataset and a map of New York city are provided by Kaggle. Originally, there are in total 15 features and 48895 rows in the dataset 'AB_NYC_2019.csv'.

After reviewing all the features, I decide to drop the features 'last_review' and 'calculated_host_listings_count', because I think the date of latest review and total amount of listing per host do not really have any effects on the prices of airbnb while people are booking. And I also decide to drop the features 'id', 'host_id', 'name' and 'host_name', because these features do not affect people whether they book this airbnb or not. What's more, I decide to drop features 'latitude' and 'longitude' as well, because people will not care about the specific location points of the airbnbs they are booking. Instead, features such as neighbourhood and neighbourhood_group provide enough geographical information. Therefore, I have 7 features in total.

Overall, the dataset is pretty well-documented. The following is a detailed description of each variable after I drop some irrelevant features.

Feature	Description (Unit)	Type
neighbourhood_group	location	categorical
neighbourhood	area	categorical
room_type	listing space type	categorical
price	price in dollar (dollar)	numerical
minimum_nights	amount of nights minimum (# nights)	numerical
number_of_review	number of reviews (# reviews)	numerical
reviews_per_month	number of reviews per month (# reviews monthly)	numerical
availability_365	number of days when listing is available for booking (# days yearly)	numerical

4. Data Preprocessing

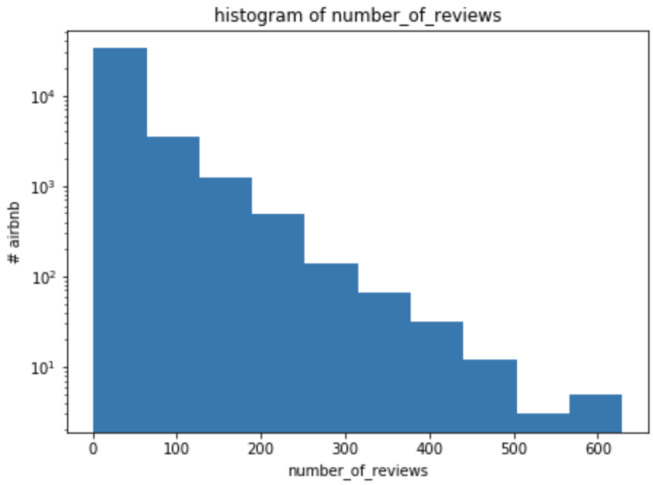
I choose to drop all the missing values for now. However, I will use some 'dropping missing value' methods to deal with the missing values later. And here I drop the features 'last_review', 'id', 'host_id', 'name', 'host_name', 'calculated_host_listings_count', 'latitude' and 'longitude'.

	neighbourhood_group	neighbourhood	room_type	price	minimum_nights	number_of_reviews	reviews_per_month	availability_365
0	Brooklyn	Kensington	Private room	149	1	9	0.21	365
1	Manhattan	Midtown	Entire home/apt	225	1	45	0.38	355
3	Brooklyn	Clinton Hill	Entire home/apt	89	1	270	4.64	194
4	Manhattan	East Harlem	Entire home/apt	80	10	9	0.10	0
5	Manhattan	Murray Hill	Entire home/apt	200	3	74	0.59	129

For features neighbourhood_group, neighbourhood and room_type, I will apply OneHotEncoder because these three features are categorical and they could not be ordered.

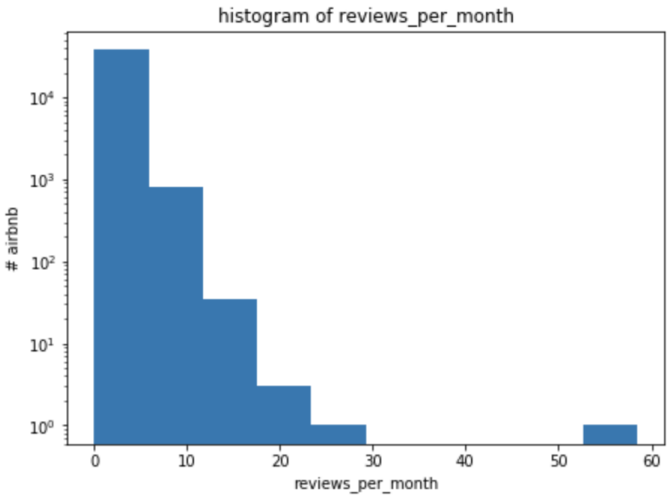
For feature minimum_nights, it is easy to notice that there are some extreme values. Therefore I decide to use StandardScaler as it is numerical.

For feature number_of_reviews, I think it is needed to plot the histogram.



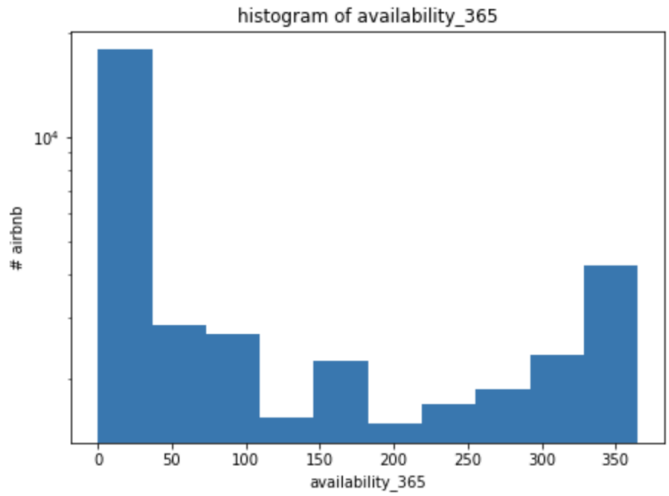
From the above histogram, I decide to use StandardScaler for the feature number_of_reviews.

For the feature reviews_per_month, I think it is needed to plot the histogram.



From the above histogram, I decide to use StandardScaler for the feature reviews_per_month.

For the feature availability_365, I think it is needed to plot the histogram.



From the above histogram, I decide to use StandardScaler for the feature availability_365.

For the target variable price, I decide to use it directly without any preprocessing, since it is continuous variable.

The final result for preprocessed-dataset is shown as follows

	neighbourhood_group_Bronx	neighbourhood_group_Brooklyn	neighbourhood_group_Manhattan	neighbourhood_group_Queens	neighbourhood
0	0.0	1.0	0.0	0.0	
1	0.0	0.0	1.0	0.0	
2	0.0	1.0	0.0	0.0	
3	0.0	0.0	1.0	0.0	
4	0.0	0.0	1.0	0.0	

5 rows × 231 columns

There are in total 7 features with 231 columns.

```
[ ]
```