New York City Airbnb calculated host listings count Prediction

Final report for DATA1030, Fall 2022 at Brown University
Junyi Zhu DSI

GitHub link: https://github.com/oooooohao/data1030project

1.Introduction

The rise of Airbnb is there for all to see. It has become a strong competitor to the traditional hotel industry. Airbnb can provide different kinds of rooms for different people. Compared to traditional hotel rooms, staying in Airbnb provides more privacy and convenience. Guests and hosts have used Airbnb to expand travel possibilities and suggest more unique, personalized ways to experience the world. (Dgomonov. 2019). At the same time, short-term rental of houses as Airbnb has also replaced long-term rental of houses as a new source of income.

Millions of listings generate vast amounts of data that can be analyzed and used for business decisions to understand customer and provider behavior and performance. The price range can be narrowed down through careful analysis of the data, making it easier to make decisions, so this project aims to analyze different characteristics and build ML models to make predictions.

I used data from insideairbnb.com. This website contains information about anything related to Airbnb in all regions of the world. It's not just about providing data for people to study, it's also a record of the economic model of Airbnb

This Airbnb dataset contains 48895 data points and 16 features. The feature description is as below. I choose the "calculated_host_listings_count" as my target variable, whose type is Int64, and plan to do a regression. The significance of predicting target variables in actual business is to predict the number of times a certain room receives tenants. Extending from this business idea, we can make a recommendation system for tenants and a ranking and scoring system for rooms.

Table 1. Feature Description

Feature Name	Type	Description
id	Int64	listing ID
name	Object	name of the listing
host_id	Int64	host ID
host_name	Object	name of the host
neighbourhood_group	Object	Location,including Brooklyn(1),Manhattan(2),Queens(3),Staten Island(4),Bronx(5) and others(6)
neighbourhood	Object	area
latitude	Float64	latitude coordinates
longitude	Float64	longitude coordinates
room_type	Object	listing space type,Entire home/apt(1),Private room(2),Shared room(3）

price	Int64	price in dollars
minimum_nights	Int64	amount of nights minimum
number_of_reviews	Int64	number of reviews
last_review	Object	latest review
reviews_per_month	Float64	number of reviews per month
caculated_host_listings_count	Int64	amount of listing per host
availability_365	Int64	number of days when listing is available for booking

Continuous Features:

- Id
- host_id
- latitude
- longitude
- price
- minimum_nights
- number_of_reviews - reviews_per_month
- calculated_host_listings_count
- availability_365

Categorical Features:

- name
- host_name
- neighbourhood_group
- neighbourhood room_type last_review

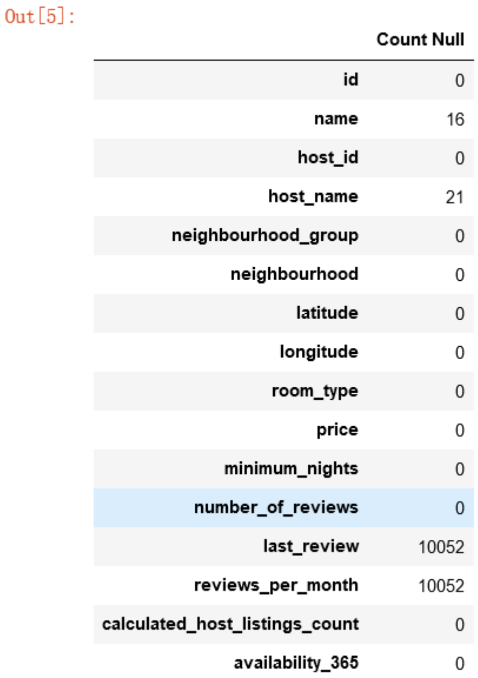
Since this dataset is very popular on Kaggle, many users have used this dataset to do some study and prediction. But most of them focused on the prediction of the price, that is, use the price as the target variable. SPUCHALSKI (2019) found that Random Forest regression model provided best accuracy for prediction of listing price based on variables generated from the initial data and the model importance can be used to further understand what drives the price of an Airbnb listing in NYC. But there is no numeric result in this research.

It is worth mentioning that the correlation of price is not as high as that of host. So this is also the reason why I choose calculated_host_listings_count as target variable.

2.Exploratory Data Analysis

2.1 Data Cleaning

The number of missing values contained in each feature is shown in the figure below:



Out[5]:

	Count Null
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

Figure 1. Missing value description

The 'last_review' and the calculated field 'reviews_per_month' have too many missing values. Considering the meaning of the field, the null value of 'reviews_per_month' can be filled with 0. Also 'last_review' can be converted to "period length from 19/12/31", and the NaN value can be set as 100000 days, which is an outlier. Since the features 'id', 'host_id', and 'host_name' are independent random variables, I decide to wash out them. The number of times the house price is 0 is 11 times. Considering that there are 48,895 pieces of data in the original data set, the 11 outliers have little impact on the dataset, so they will not be processed.

2.2 EDA on each column

For the target variable 'calculated_host_listings_count', I plot the histogram with logarithmic scale to clearly show the target variable distribution. Figure 2 below shows that there is a right tail distribution of the target variable. Most of them are between 1 and 50. By using the describe function to see the details of each column in the dataset, where the minimum value of the target variable ‘calculated_host_listings_count’ is 1 and the maximum value is 327.

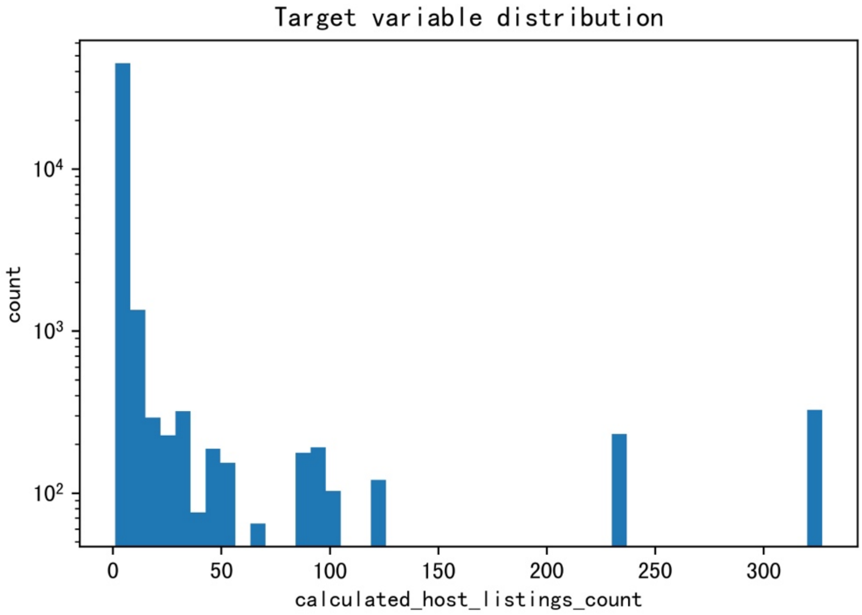


Figure 2. Target variable distribution with Log Scaling

2.3 EDA on two columns and Pearson matrix

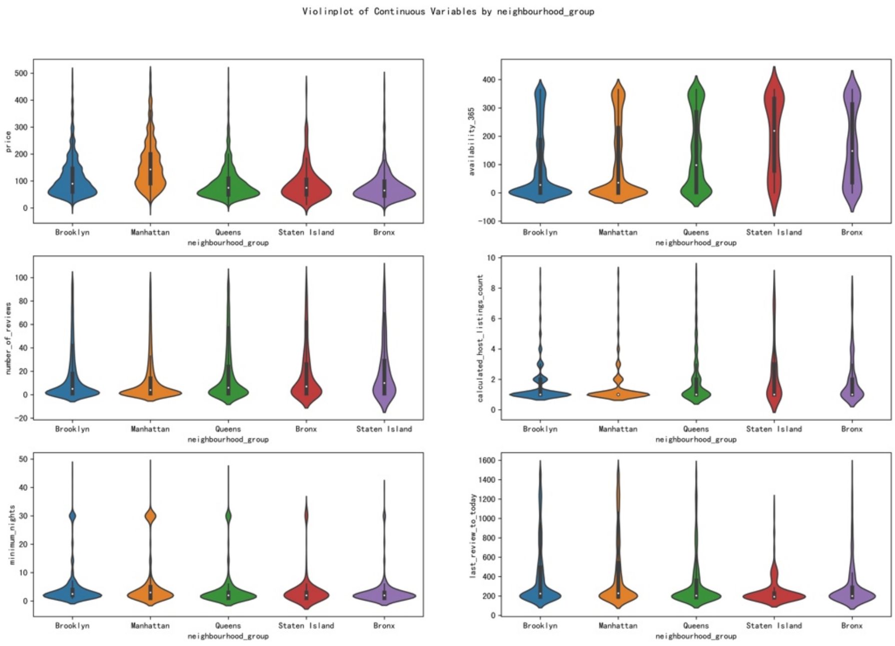


Figure 3. Distribution of Price by neighbourhood_group

As shown in Figure 3 above, the price of Manhattan does not show a centralized trend, and each range is evenly distributed. But the prices of other districts are concentrated at around 60. The availability_365 shows more distribution at both ends and less distribution in the middle. The distribution of Brooklyn and Manhattan is similar, which concentrates around 0 days. It is speculated that the Airbnb registrations in these two districts are mainly self-occupied houses. The other three districts have similar distributions at the beginning and the end, and the proportion of self-occupied and rental houses is close. The number_of_reviews still shows a similar distribution in Brooklyn and Manhattan. The number of house reviews is mostly concentrated between 0 and 10. In areas with fewer houses, it is more likely that the distribution of comments will be concentrated. The calculated_host_listings_count still shows a similar distribution in Brooklyn and Manhattan. In these two areas, users have concentrated orders for 1 or 2 times. It is speculated that these two areas are tourist areas, and there are more one-time tourist tenants. The distribution of minimum_nights in various districts is similar, mainly concentrated in 0-10 days, but the Manhattan district has a distribution of about 30. The last_review_to_today is derived from the last_review feature, where the null value is set by us as the default value of 100000. From the distribution point of view, the last comment time is concentrated about 200 days ago, that is, around 2019/06. Staten Island has the most concentrated distribution with a maximum of about 1200, that is, around 2016/09. Comments have already appeared in other areas. It is speculated that Airbnb has fewer users in this area, or there is less demand for short-term rentals.

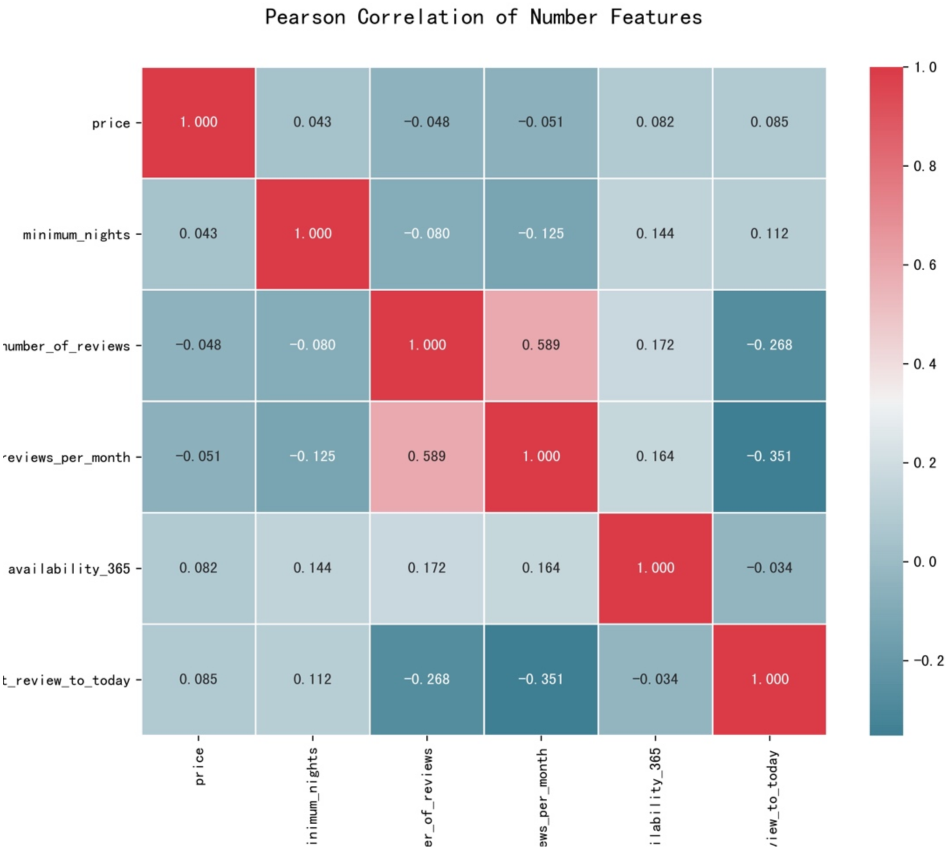


Figure 4. Pearson Correlation of Continuous Features

As can be seen from the heatmap above, there is no strong correlation here, which is a good news for machine learning. The number_of_review and the reviews_per_month have the strongest positive correlation. The Last_review_to_today and the reviews_per_month have the strongest negative correlation.

3.Methods

3.1Data Splitting and Preprocessing

This dataset is IID which is independent and identically distributed. Each data point is independent, but they satisfy the same distribution. This dataset also has no group structure because all samples are selected in the same way.

Breaking the dataset into training sets can help us understand the model, which is important for how the model generalizes to new unseen data. If the model is overfitted it may not generalize well to new unseen data. Therefore, it is hard to make good predictions.

In general, we are used to using 80% of the original data as the training set, 20% of the original data as the test set. Since this dataset is IID and a big dataset, I would like to use the basic split to split the dataset with test size is 20%, then use KFold split with the number of split 4 for the other 80% of the data. In this way, the mean and variance of the training set and test set are close, which can better prove the generality of the model. Also taking into account the balance of the dataset as well as the practical significance of the target variable, taking the above split data helps to find the best hyperparameters for my ML algorithm and shows how the model will handle previously unseen data by applying it to the final set.

Then I decide what scalers and encoders to use based on the type of different features. I perform one-hot encoding for the categorical features because these three categories ('neighbourhood_group','neighbourhood','room_type') cannot be sorted. For the feature value 'room type', I use ordinal encoding because they have a natural rank ordering. The violin chart in the EDA analysis above shows that most of the continuous features follow the tail distribution, so StandardScaler is better to be used here. Also, there are no reasonable boundaries, so it is not appropriate to use minmax here.

After the above processing, I have the training, validation, and test groups. There are 11 features in the dataset, and the number of data points is 48895.

3.2 Model Selection

Using the segmentation and preprocessing strategies mentioned in Section 3.1, I trained 7 machine learning models: Linear Regression with three types of regularization (L1, L2, Elastic Net), Random Forest, Support Vector Regression, K-Neighbors Regression and XGBoost regression.

All these models were hyperparameter tuned to return the best model and test scores, and the test was repeated across 10 random states to measure the variance and uncertainty of the model predictions. The name of each model, tuned parameters and tested values are as follows.

Table1. Parameters used for tuning of each model

Name of Models	Parameters
L1 regularized linear regression	'alpha': np.logspace(-7,0,5)
L2 regularized linear regression	'alpha': np.logspace(-7,0,5)
linear regression with an elastic net	'alpha': np.logspace(-7,0,5)

	'l1_ratio': [0.1,0.2,0.3]
Random Forest	'max_depth': [1,5,10,30] 'min_samples_split': [2,6,10,16]
Support Vector Regression	'C':[0.001, 0.01, 0.1, 1, 10, 100,1000] 'gamma': [0.001, 0.01, 0.1, 1, 10, 100, 1000]
K-Neighbors regression	'n_neighbors': [1,10,30,100]
XGBoost regression	'reg_alpha': [0e0, 1e-2, 1e-1, 1e0, 1e1, 1e2] 'reg_lambda': [0e0, 1e-2, 1e-1, 1e0, 1e1, 1e2] 'max_depth': [1,3,10,30,100]

The metric I use to evaluate the performance of my models is Root Mean Squared Error (RMSE) because it penalizes large prediction errors while Mean Absolute Error (MAE) does not. In addition, RMSE keeps the dimension of the estimated value consistent with the dimension of the original value so it is better than the mean square error (MSE). Because MSE simply and straightforwardly expresses the prediction error, but the square is added to expand the error value, and the evaluation result is unclear when the dimension is large. For this reason, when training a model as a loss function, the model is heavily influenced by outliers.The uncertainty caused by splitting is that if the seed or random variable changes, the outcome will change. There is also an uncertainty due to the non-deterministic ML methods I used Random Forest and XGBoost, which are sensitive models and produce variable results. (Gupta . 2021)

4.Result

Choose the baseline as the mean of the target values. The baseline model returned an average RMSE score of 202.56 with a standard deviation of 4.67.

Of all the models, the random forest model was the most predictive, and the performance of the ML model including the pipeline is shown in the figure below. The average RMSE score for the random forest model is 13.69 with a standard deviation of 0.849.

This is followed by the KNN model with an average RMSE score of 19.08 and a standard deviation of 0.740.

Using the linear regression model of L1 regularization, L2 regularization and Elastic Net, the average RMSE score of these three models is around 28, and the standard deviation is around 1.04.

The worst predictive performance is the SVR model with an average RMSE score of 45.23 and a standard deviation of 2.48.

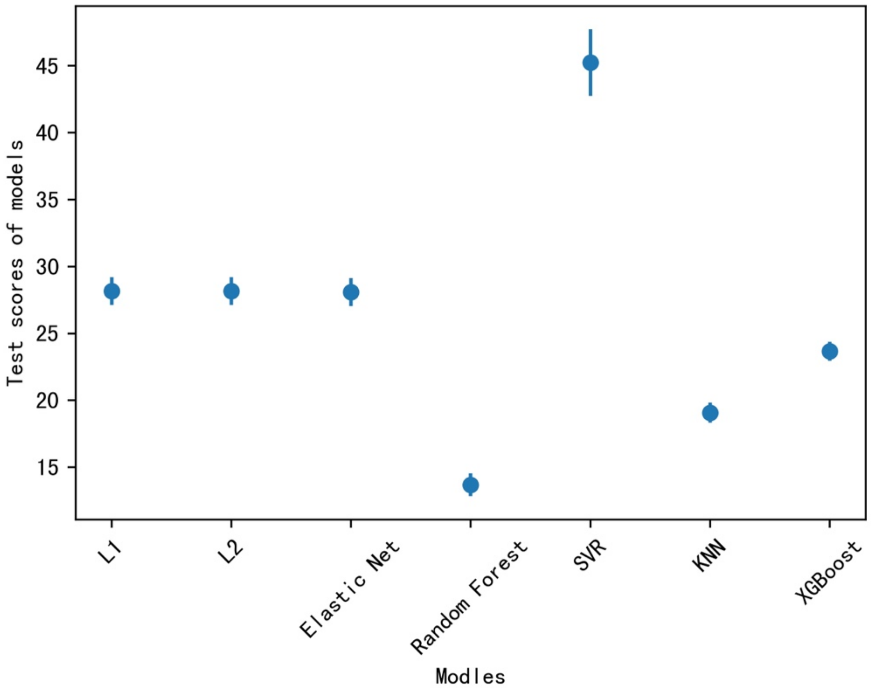


Figure 1. RMSE scores for each model

In addition, I also calculated the importance of different global features of the random forest, using permutation, mean decrease in impurity. All these show high similar results, that is, availability_365, minimum_nights, price, and listing location are relatively important. For other features, their importance is relatively low.

I also calculated the SHAP value for local feature importance, and the results show that the local importance calculated using the SHAP method is highly similar to the three global importance results.

According to the figure below, it is unexpected that the importance of room_type and neighbor_group is so low. The fact is that the resources in different neighbor_group are extremely unbalanced, so housing prices in different neighbor_group and different room_types are different, which will lead to differences in the number of rented houses. But my model results show that region is not one of the important factors affecting the target variable, but its latitude and longitude does have an important impact on it.

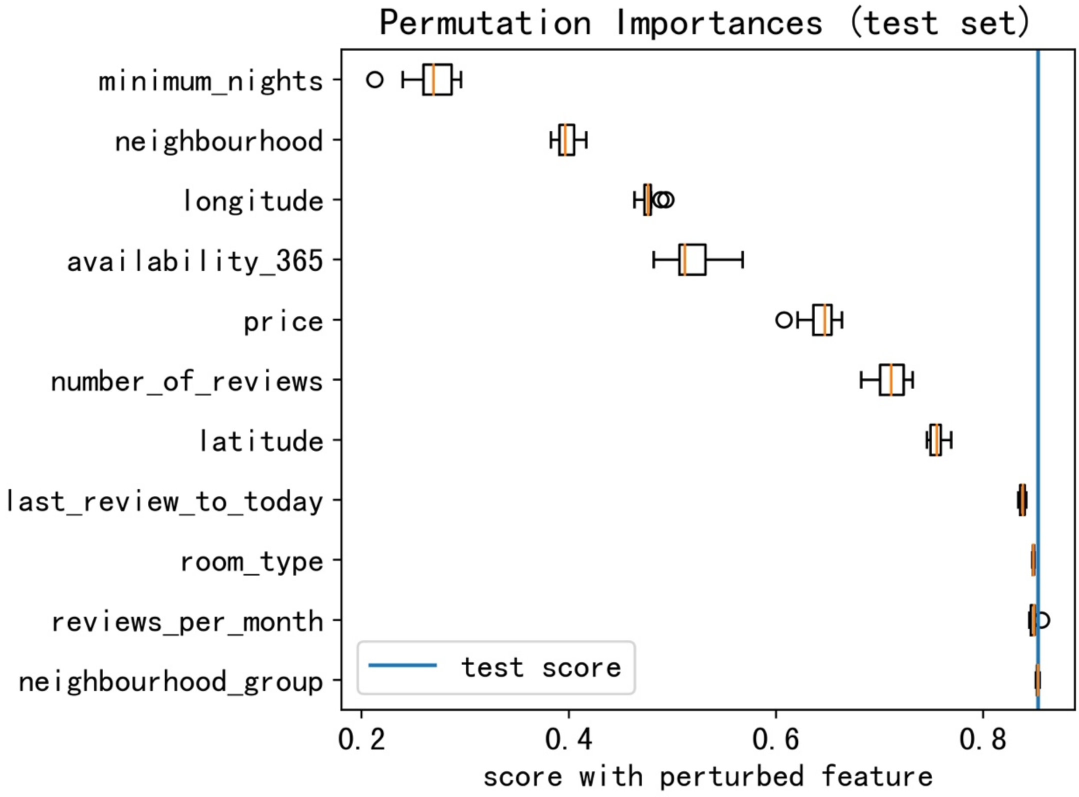


Figure 2. Permutation Importances for features from the Random Forest Model

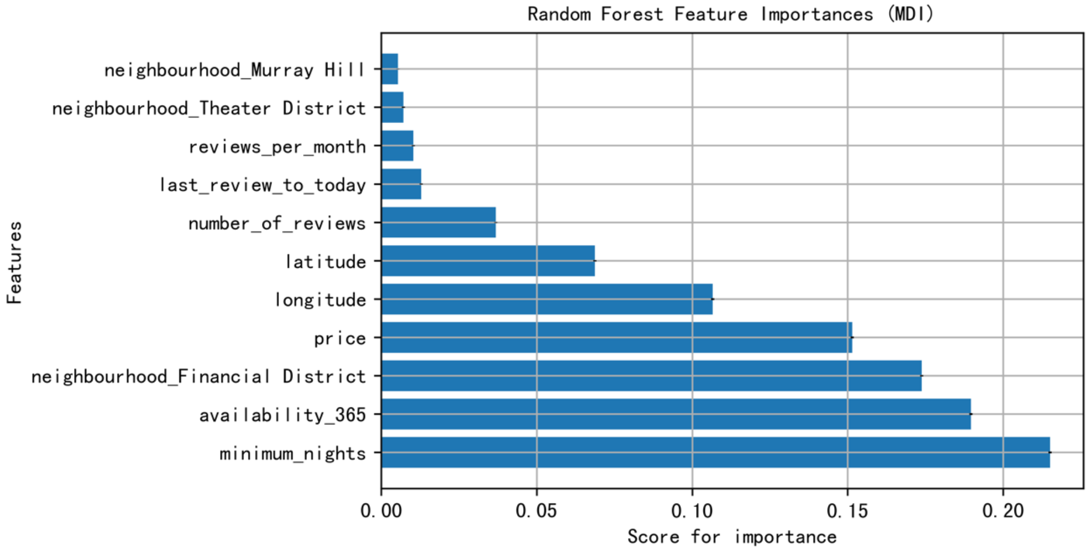


Figure 3. Random Forest Feature Importances (MDI)

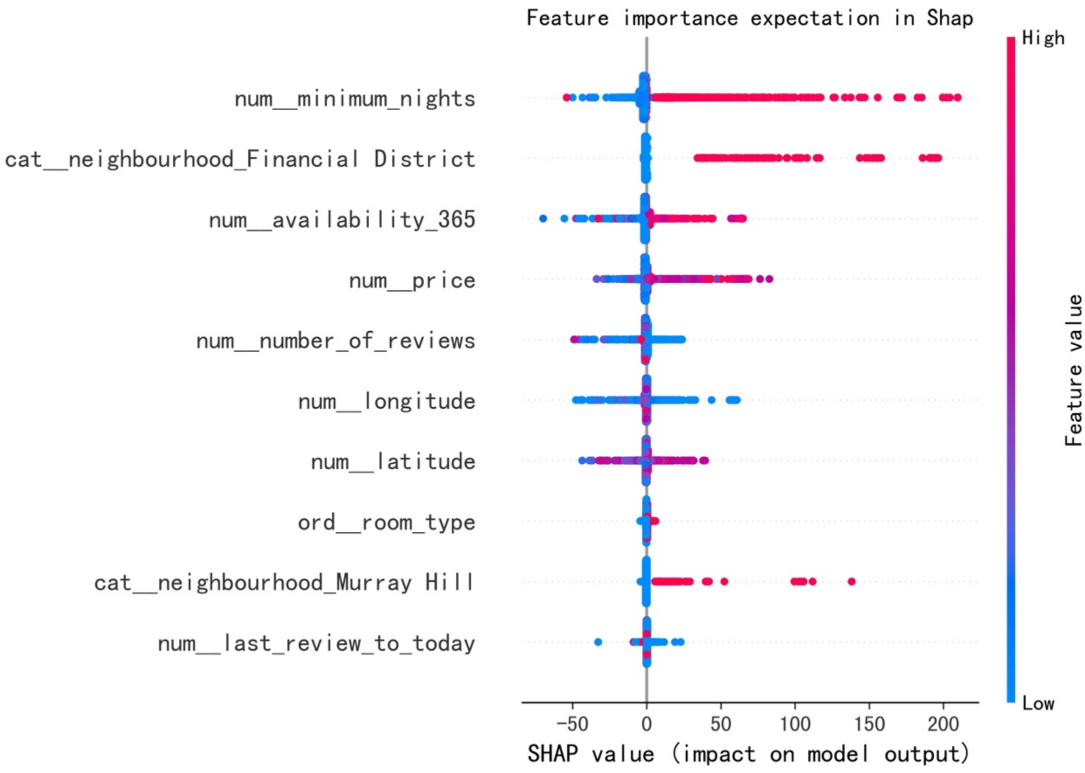


Figure 3. SHAP value for top 10 features from the Random Forest Model

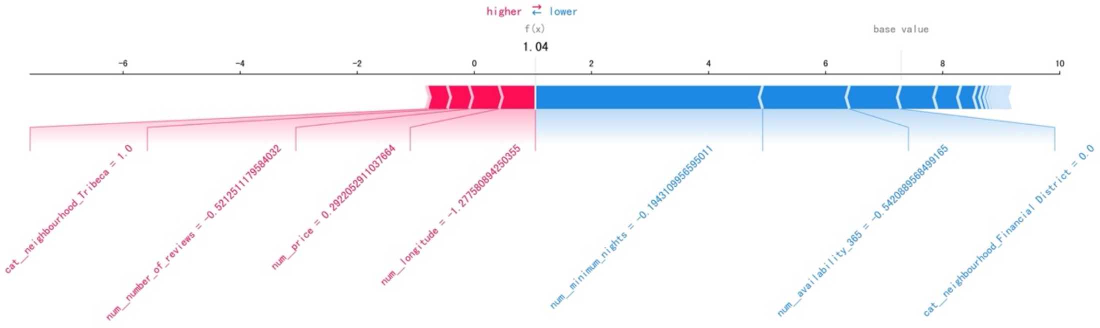


Figure 4. SHAP value for local importance

5.Outlook

The results show that the number of days when listing is available for booking, the price of the house, and the location all determine the amount of listing per host. This makes sense, since these are characteristics that people typically consider when booking a room.

In addition, the model cannot explain the importance of features such as the area where the house is located. It may be that only some valuable features are retained for analysis when performing feature selection, and the features of longitude and latitude may have some overlapping information with neighbor and neighbor_group.

Other techniques I can use for this regression problem are decision tree regression and neutral network regression. Additional data, such as the area of the house, whether the owner has other jobs, the floor of the house, whether the user will stay again, etc. can be collected to demonstrate the performance of the model.

6.Reference

Dgomonov. (2019, August 12). New York City airbnb open data. Kaggle. Retrieved October 21, 2022, from https://www.kaggle.com/datasets/dgomonov/new-york-city-airbnb-open-data

Spuchalski. (2019, October 3).Predicting price of airbnb listings in NYC. Kaggle. Retrieved October 21, 2022, from https://www.kaggle.com/code/spuchalski/predicting-price-of-airbnb-listings-in-nyc

Gupta, A. (2021, June 1). XGBoost versus Random Forest. Medium. Retrieved December 5, 2022, from https://medium.com/geekculture/xgboost-versus-random-forest-898e42870f30

GitHub link: https://github.com/oooooohao/data1030project