How much should I charge?

Airbnb Listing Price Prediction using Machine Learning

Kimberly Hu

2024-04-29

Abstract

Setting the right price for an Airbnb rental is a key challenge faced every Airbnb host. In this project, I develop a predictive model to recommend listing prices to hosts, utilizing Airbnb listing data in New York City from October 2023 to March 2024, enriched with local points of interest data. I employed multiple algorithmic approaches, and identified that the XGBoost model outperformed others, achieving a minimum RMSE of approximately 67. An examination of feature importance reveals that factors such as the accommodation capacity, proximity to the city center, minimum stay requirements, and the number of bedrooms and bathrooms critically influence pricing decisions. Although the model exhibits limitations in its predictive accuracy, it provides valuable insights for hosts in setting rental prices. Several enhancements can be made to improve the performance of the model, including integrating historical booking data, applying time series analysis, and further model optimization.

1 Introduction

Setting an appropriate price for an Airbnb listing is crucial for success in the business, yet determining the optimal price can be quite complex. The listing price may be influenced by a wide variety of factors, including both the inherent characteristics of the apartment and the host, as well as external seasonal and market influences. Seasonal factors, such as holidays and local events, induce price fluctuations across the entire market, whereas the specific characteristics of the apartment and the host drive differences among individual listing prices. Since the aggregate market demand is hard to forecast, this project focuses on modeling prices based on the listing characteristics to provide hosts with insights into their relative position in the market.

Predicting Airbnb prices is of interest to many people, and there are numerous machine learning based projects on this topic. However, many of them rely solely on numerical or categorical data scraped from Airbnb, and do not incorporate data from additional sources. This project distinguishes itself by integrating extra information through extensive feature engineering, such as extracting keywords from titles and descriptions, calculating time differences between dates, and measuring distances between the property and specific sites. These steps enrich the data set and is beneficial for building a model with higher predictive accuracy. In addition, the project focuses specifically on New York City, due to its highly active Airbnb market and diverse properties. Since the average level of price varies greatly across different cities, it is more reliable to compare property prices within the same city.

2 Methods

Data

The main data of this project comes from Inside Airbnb, which is an open data project that shares real listing information scraped from the Airbnb website. The data set I used is the “detailed listings data” for

New York City and contains information related to listing and host. I merged the monthly detailed listings data from October 2023 to March 2024 and kept distinct listings to form a full data set containing 46403 observations.

Point of interest data are collected from NYC Open Data. The data set is compiled by multiple city agencies. I extracted the coordinates of Central Park, the Empire State Building, and the Downtown Financial District from the data set.

Feature engineering

Based on the collected data, I created some extra features for the listings:

- **Amenities:** Each Airbnb listing features a list of amenities associated with it. I examined the frequencies of these amenities across all listing data and selected five that appear with medium frequencies and might impact a guest's choice to stay. Subsequently, I created dummy variables to indicate whether an Airbnb listing includes these amenities.
- **Days between scraped time and review time:** I calculated the number of days between the scrape time and the first review, as well as the number of days between the scrape time and the last review. These indicate how long ago the reviews were made.
- **Keywords from listing title and description:** I extracted commonly used keywords from the listing title and listing description respectively, then created dummy variables for whether the title or description contained those keywords.
- **Length of neighborhood and host descriptions:** The length of the neighborhood description and the host description as used as measures of complexity of these texts.
- **Distance to sites:** I calculated the distances between the property and the Central Park, the Empire State Building, and the Downtown Financial District, respectively. These sites were chosen because they are of interest to both tourists and residents of the city.

The full set of features are described in the appendix.

Model building and evaluation

After eliminating highly correlated variables and scaling numerical data, I built four types of models and evaluated their performance. The measure of prediction accuracy is out-of-sample root mean squared error (RMSE). This metric evaluates how far off the predictions are from the actual values. A model with lower RMSE has better performance.

- **LASSO regression:** LASSO is a regularization technique used over regression in order to enhance the prediction accuracy and interpretability of the model. The algorithm reduces some coefficients to zero, thus performing automatic feature selection, and is especially useful when there is a large number of features. The lambda value in a LASSO model determines the amount of regularization. Larger lambda leads to more coefficients being pushed towards zero. I used cross-validation to select the best lambda value which minimizes cross-validation error, then built a LASSO regression model using this lambda value.
- **Random forest:** Random forest utilizes both bagging and feature randomness to create an uncorrelated forest of decision trees. The randomness introduced by the algorithm reduces the risk of overfitting by a single decision tree. I built a random forest model using the default parameters, which I found worked better than other parameters.

- **XGBoost:** XGBoost is an implementation of gradient boosted trees. Gradient boosting attempts to predict a target variable by combining the estimates of a set of simpler, weaker models. XBoost is computationally efficient and offers regularization to prevent overfitting, making it suitable for large data sets. Since the model is sensitive to its parameters, I found the best performing model after tuning several hyperparameters.
- **Quantile regression forest:** Combining quantile regression with random forest, the quantile regression forest estimates the selected quantile rather than the mean prediction from the trees. It can be used to predict the median, which is more robust to outliers in the data than the mean. In this project, I built a quantile regression forest using the default parameters.

3 Results

Exploratory analysis

Although the whole data set contains 46,403 observations, some of them are missing prices and removed. Further inspection shows that the top and bottom prices may be invalid. There are listings above 10,000 dollars per night and listings below 20 dollars per night, but their characteristics do not match with these abnormal prices. Thus, I removed observations with the top and the bottom prices to filter out invalid listings. The total number of observations left is 29,343.

Figure 1 plots the listings on a map of NYC, colored by prices. Airbnb rentals are scattered across the 5 boroughs of NYC, and there are high prices and low prices in each borough. However, Midtown and Lower Manhattan tend to have more high priced apartments than other areas.

Figure 2 shows the price of the listings by room type. As expected, entire apartments have the highest price on average, and the price range of private rooms and shared rooms are closer. For each room type, there are still some listings with significantly higher prices, even after removing the outliers.

Table 1 shows the listings by the minimum number of nights required for booking. Most of the listings requires a minimum of 30 nights, likely due to a restriction on short-term rentals in NYC that came into effect in September 2023. I also noticed that prices tend to be lower for longer term rentals, which makes sense since hosts generally give reductions when the rental term is long.

Due to the page limit, additional exploratory analysis are in the appendix.

Table 1: Price by minimum stay requirement

Minimum number of nights	Count	Average price	Median price
Less than a week	4310	236.79	198.5
One week to one month	248	231.52	181.0
One month to three months	24315	156.35	122.0
More than three months	470	122.21	99.0

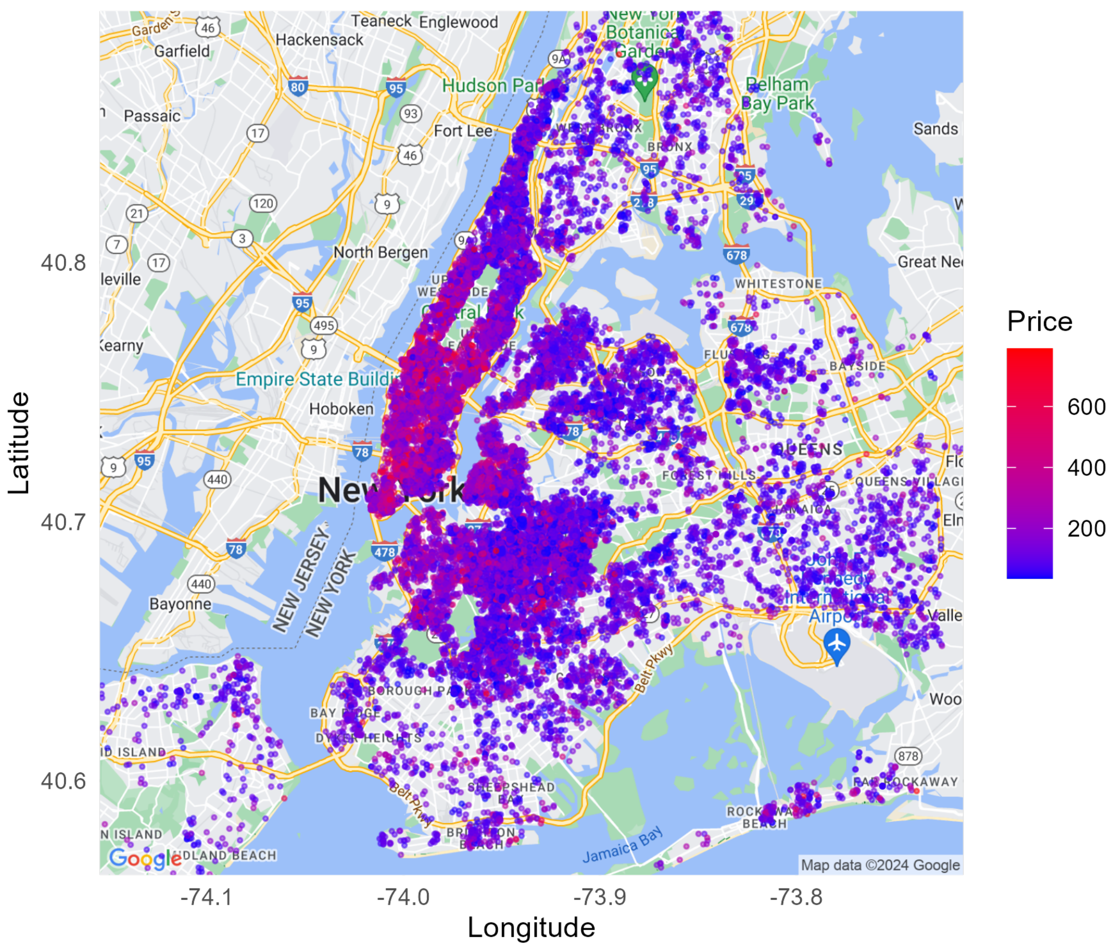

Figure 1: Map of Airbnb listings in NYC

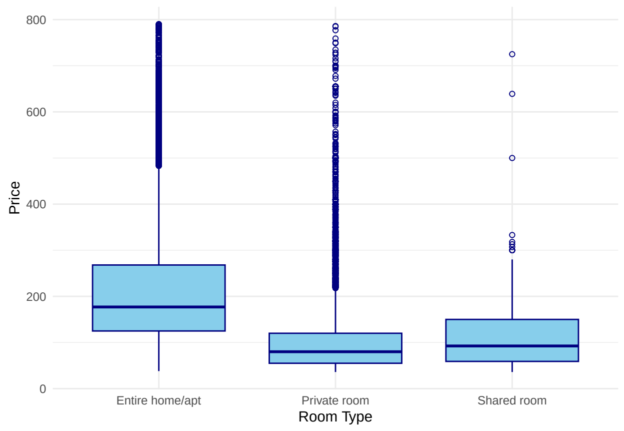

Figure 2: Price by room type

Model evaluation

Before building the models, I first checked the correlation between the variables to avoid multicollinearity problem, by calculating the variance inflation factors (results shown in the appendix). `dist_park`, `dist_dt`, and `review_scores_rating` are excluded from the set of predictors because they are highly correlated with other variables. Out of the three distance features, I chose to keep the distance to the Empire State Building, because the building not only serves as a top tourist site, but is also located at the center of the city. In addition, observations with NA values are removed, because NA values cannot be used in regression and random forest. All numerical variables are scaled.

Table 2 shows the RMSE values of the predictive models. Lower RMSE values indicate higher out-of-sample prediction accuracy. By comparison, XGBoost outperforms the other models by achieving an RMSE value of 67.43. Thus, I selected it as the model for price prediction.

Table 2: RMSE of Predictive Models

Model	RMSE
LASSO regression	84.51431
Random forest	69.86460
XGBoost	67.43793
Quantile regression forest	71.53155

Feature importance

Base on the XGBoost model, the top features that drive variation in price are the number of guests the property accommodates, the distance to Empire State building, the number of bedrooms and bathrooms, and the minimum number of nights required for booking. The results show that size of the apartment, location, and rental term are the most important things to consider when determining price.

Figure 4 presents the partial dependency plots of the important features. Price increases as the size of the property increases. Price decreases with further distance to city center and longer rental period.

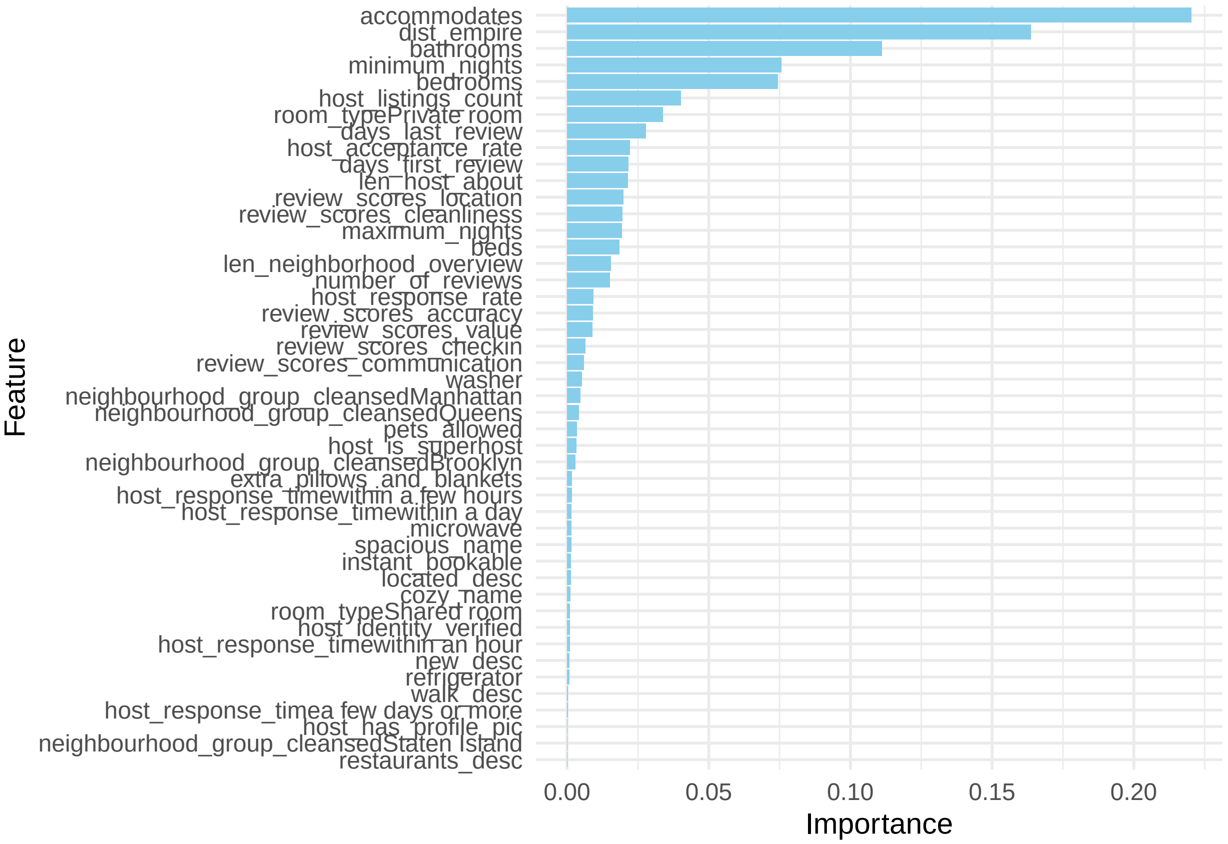

Figure 3: Feature importance in XGBoost model

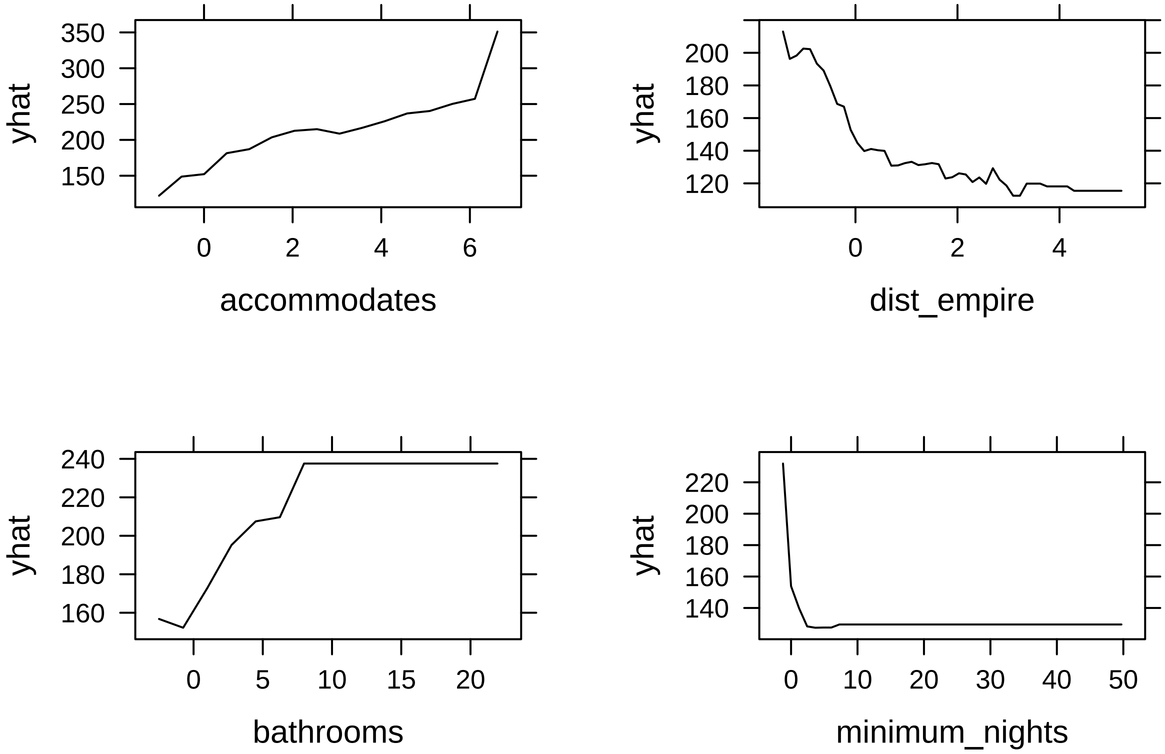

Figure 4: Partial dependency plots

4 Conclusion

Trained on the listing characteristics of Airbnb rentals, an XGBoost model outperforms other models by accurately predicting the listing price with an RMSE of 67.43. Important features that drive variation in price include the accommodation capacity, proximity to the city center, minimum stay requirements, and the number of bedrooms and bathrooms. Prices are higher when the apartment is large and close to city center. Since the Airbnb rentals in New York City are mostly long-term rentals with minimum stay requirement of longer than 30 days, the characteristics that affect prices in New York City may be different from those in other cities. Using listing characteristics and the XGBoost model, an Airbnb host can get a sense of whether their pricing is appropriate.

The approach of this project is limited by data access and resources, and several adjustments can potentially enhance the performance of the model. First, I do not have access to historical booking prices, only listed prices scraped monthly from the Airbnb website. This could introduce bias into the model, since the listed prices are not equivalent to booked prices. Some apartments in the data set may have never been booked at the listed price. Thus, training the model on the listing price may not provide accurate predictions of actual price. Second, price trend over time can significantly affect the pricing strategy of hosts. In this project, I did not consider change in the price of a listing over time, since it is complex to model market demand and seasonal factors. However, adding time series analysis can potentially increase the accuracy of the predictions. Finally, due to limited computational capacity, I did not perform extensive search on the optimal hyperparameters of the model. With resources and time permitting, a grid search or random search can be used in tuning a better model. Treating the NA values as it is, rather than omitting them, can also provide extra information for model training.

Appendix

Table 3: Variable description

Name	Description
price	Price of listing (target variable)
host_response_time	Category for host response time
host_response_rate	Rate at which a host responds to booking requests
host_acceptance_rate	Rate at which a host accepts booking requests
host_is_superhost	Indictor variable of whether the host is a superhost
host_listings_count	Total number of listings of the host
host_has_profile_pic	Indictor variable of whether the host has a profile picture
host_identity_verified	Indictor variable of whether the host verified his/her identity
neighbourhood_group_cleansed	The borough that the property is located in
room_type	Category for room type
accommodates	Maximum number of guests that the property accommodates
bathrooms	Number of bathrooms; half bath counted
bedrooms	Number of bedrooms
beds	Number of beds
minimum_nights	Minimum number of nights required for booking
maximum_nights	Maximum number of nights allowed for booking
number_of_reviews	Number of reviews
review_scores_rating	Overall rating (1-5)
review_scores_accuracy	Rating for accuracy (1-5)
review_scores_cleanliness	Rating for cleanliness (1-5)
review_scores_checkin	Rating for check-in (1-5)
review_scores_communication	Rating for communication (1-5)
review_scores_location	Rating for location (1-5)
review_scores_value	Rating for value (1-5)
instant_bookable	Indicator variable of whether the guest can automatically book the listing
refrigerator	Indicator variable of whether the property has a refrigerator
microwave	Indicator variable of whether the property has a microwave
washer	Indicator variable of whether the property has a washer
pets_allowed	Indicator variable of whether pets are allowed on property
extra_pillows_and_blankets	Indicator variable of whether the property has extra pillows and blankets
days_first_review	Number of days since the first review
days_last_review	Number of days since the last review
len_neighborhood_overview	Character length of neighborhood description
len_host_about	Character length of host description
cozy_name	Indicator variable of whether the title of the listing contains "cozy"
spacious_name	Indicator variable of whether the title of the listing contains "spacious"
located_desc	Indicator variable of whether the description of the listing contains "located"
restaurants_desc	Indicator variable of whether the description of the listing contains "restaurants"
walk_desc	Indicator variable of whether the description of the listing contains "walk"
new_desc	Indicator variable of whether the description of the listing contains "new"
dist_empire	Distance to the Empire State Building
dist_dt	Distance to the Central Park
dist_park	Distance to the Downtown Financial District

Table 4: Count of listings by borough. Most Airbnb rentals are located in Manhattan or Brooklyn, but there is a fair amount of apartments in each borough.

Borough	Count
Bronx	1258
Brooklyn	10312
Manhattan	12742
Queens	4672
Staten Island	359

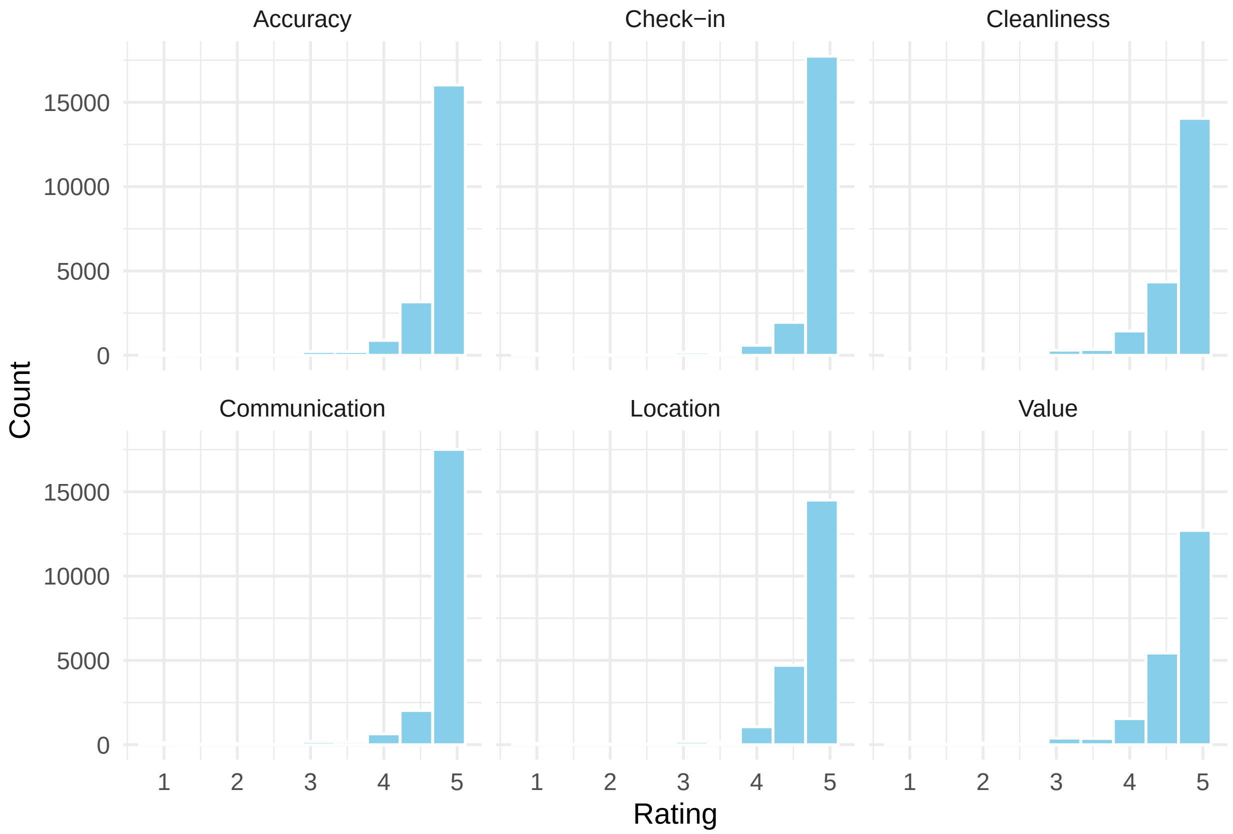

Figure 5: Distribution of ratings. Ratings for each aspect of the apartments are distrbuted similarly, with most ratings at 5 and few ratings below 3.

Table 5: Variance inflation factor (VIF) analysis

	GVIF	Df	GVIF^(1/(2*Df))
host_response_time	5.778968	3	1.339600
host_response_rate	4.953092	1	2.225554
host_acceptance_rate	1.587411	1	1.259925
host_is_superhost	1.259245	1	1.122161
host_listings_count	1.277519	1	1.130274
host_has_profile_pic	1.025693	1	1.012765
host_identity_verified	1.046324	1	1.022900
neighbourhood_group_cleansed	5.800748	4	1.245763
room_type	1.539203	2	1.113843
accommodates	3.060178	1	1.749337
bathrooms	1.284700	1	1.133446
bedrooms	2.116638	1	1.454867
beds	2.635459	1	1.623410
minimum_nights	1.172970	1	1.083037
maximum_nights	1.070698	1	1.034745
number_of_reviews	1.491888	1	1.221429
review_scores_rating	7.283238	1	2.698747
review_scores_accuracy	4.378571	1	2.092504
review_scores_cleanliness	2.897269	1	1.702136
review_scores_checkin	2.571747	1	1.603667
review_scores_communication	3.464295	1	1.861262
review_scores_location	1.740189	1	1.319162
review_scores_value	4.125189	1	2.031056
instant_bookable	1.260660	1	1.122791
refrigerator	1.285983	1	1.134012
microwave	1.395995	1	1.181522
washer	1.104146	1	1.050783
pets_allowed	1.108187	1	1.052705
extra_pillows_and_blankets	1.180923	1	1.086703
days_first_review	1.668605	1	1.291745
days_last_review	1.367735	1	1.169502
len_neighborhood_overview	1.160446	1	1.077240
len_host_about	1.216975	1	1.103166
cozy_name	1.025545	1	1.012692
spacious_name	1.015027	1	1.007485
located_desc	1.003119	1	1.001558
restaurants_desc	1.003251	1	1.001624
walk_desc	1.020104	1	1.010002
new_desc	1.036362	1	1.018019
dist_dt	15.006583	1	3.873833
dist_park	16.349877	1	4.043498
dist_empire	40.207998	1	6.340978