# Predicting AirBnB Rental Price in NYC

Jisu Baek

## 1. Overview

The goal of the project is to predict the Airbnb rental price precisely. Airbnb data from the New York City (NYC) in 2018 was used to predict the rental price and identify factors that influence Airbnb prices. The dataset consists of 29,142 observations with 96 features. Proper pre-processing and Exploratory Data Analysis (EDA) will be performed using univariate analysis and Least Absolute Shrinkage and Selection Operator (LASSO) to select meaningful predictors when modeling the log-transformed price. Models using Multiple Linear Regression, Decision Tree with tuning, Random Forest, and Boosting Model will be utilized to predict the rental prices and the performance of each model will be evaluated by Root Mean Square Error (RMSE).

## 2. Preprocessing Data and Exploratory Data Analysis

In the given dataset, train and test was divided into 8:2 ratio (train set: 23313 observations; test set: 5829 observations). Out of 96 variables, 54 variables were excluded from the analysis as they are not informative to predict the price, or there are too many missing values. Variables with >20% NAs were excluded from the analysis. Some categories with small observations were recategorized to minimize the bias in prediction. However, some variables, such as cleaning fee, the NA values were imputed with $0. All the variables with True/False were converted into 1/0 as numeric variables. Amenities converted into `amenity_num` by counting the number of amenities for each listing. 14 observations with missing outcome variables were excluded from the analysis as well. [Figure 1] showed the complete code and the list of variables that were excluded from the analysis. Before and after the preprocessing, the relationship among variables were visually checked using GGally::ggpairs() function.

```
train <- train %>% dplyr::select(-listing_url, -scrape_id, -last_scraped, -name, -summary, -space, -description, -experiences_offered, -neighborhood_overview, -notes, -transit, -access, -interaction, -house_rules, -thumbnail_url, -medium_url, -picture_url, -xl_picture_url, -host_id, -host_url, -host_name, -host_since, -host_location, -host_about, -host_acceptance_rate, -host_thumbnail_url, -host_picture_url, -host_neighbourhood, -host_verifications, -street, -neighbourhood, -city, -state, -zipcode, -market, -smart_location, -country_code, -country, -square_feet, -weekly_price, -monthly_price, -security_deposit, -cleaning_fee, -calendar_updated, -calendar_last_scraped, -first_review, -last_review, -license, -jurisdiction_names, -host_response_time, -host_response_rate, -neighbourhood_cleansed, -requires_license, -has_availability)
```

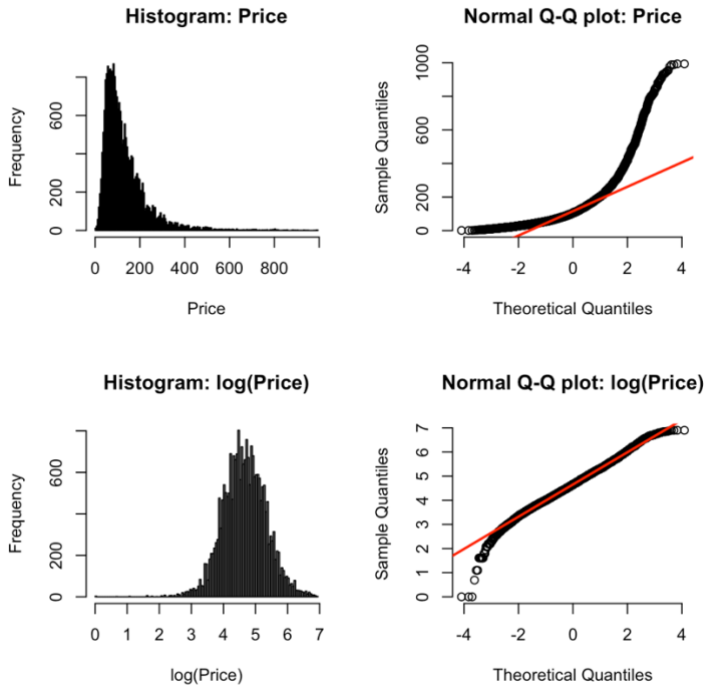
[Figure 1] The code (list) of variables that were excluded from the analysis.

## 3. Feature Selection

1) Log-transformation of outcome variable, price

   Before we select meaningful features to predict rental prices, log-transformation was made on the outcome variable, price. According to [Figure2], As the original outcome

variable were skewed, transformation was considered. Among many transformations, it turned out that log-transformation was quite effective to correct this issue.




[Figure 2] Histogram and Normal Q-Q plots of price before and after log-transformation

## 2) Univariate analysis

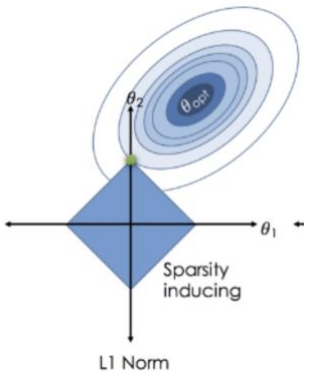
With each independent variable, log-transformed price was paired to check the associations. To identify the associations, two-sample t-test and ANOVA test was used between log(Price) and categorical variables. Between log(Price) and continuous variables, Pearson's correlation (*r*) was used. [Figure 3] demonstrated that multiple variables were associated with the log-transformed price.

| Variables | pval | correlation |
|---|---|---|
| accommodates | <0.001 | 0.538 |
| beds | <0.001 | 0.424 |
| guests_included | <0.001 | 0.357 |
| bedrooms | <0.001 | 0.345 |
| review_scores_location | <0.001 | 0.202 |
| amenities_num | <0.001 | 0.182 |
| bathrooms | <0.001 | 0.137 |
| extra_people | <0.001 | 0.126 |
| review_scores_cleanliness | <0.001 | 0.094 |
| review_scores_rating | <0.001 | 0.078 |
| latitude | <0.001 | 0.073 |
| review_scores_accuracy | <0.001 | 0.050 |
| review_scores_communication | <0.001 | 0.042 |
| review_scores_checkin | <0.001 | 0.031 |
| availability_365 | <0.001 | 0.024 |
| number_of_reviews | 0.003 | 0.019 |
| availability_30 | <0.001 | -0.026 |
| reviews_per_month | <0.001 | -0.044 |
| availability_60 | <0.001 | -0.048 |
| availability_90 | <0.001 | -0.057 |
| calculated_host_listings_count | <0.001 | -0.157 |
| longitude | <0.001 | -0.339 |
| host_is_superhost | 0.028 | |
| host_identity_verified | <0.001 | |
| neighbourhood_group_cleansed | <0.001 | |
| is_location_exact | <0.001 | |
| property_type | <0.001 | |
| room_type | <0.001 | |
| bed_type | <0.001 | |
| instant_bookable | <0.001 | |
| is_business_travel_ready | <0.001 | |
| cancellation_policy | <0.001 | |
| require_guest_phone_verification | 0.013 | |

[Figure 3] Univariate analysis result

### 3) LASSO

The LASSO method was also utilized for feature selection process. LASSO regularizes model parameters by shrinking the regression coefficients using L1 regularization. Also, cross-validation was included in this process to choose the penalty factor.




[Figure 4] LASSO L1 regularization visualization

After the univariate analysis and LASSO feature selection process, 23 features were selected as meaningful variables to predict the price. These variables will be used to generate prediction models. Here is the list of the significant variables:

```
host_is_superhost, host_identity_verified, neighbourhood_group_cleansed,
longitude, property_type, room_type, accommodates, bathrooms, bedrooms,
guests_included, extra_people, minimum_nights, availability_30, availability_365,
review_scores_rating, review_scores_cleanliness, review_scores_location,
review_scores_value, instant_bookable, is_business_travel_ready,
calculated_host_listings_count, reviews_per_month, amenities_num
```

[Figure 5] Final 23 features associated with the log-transformed price

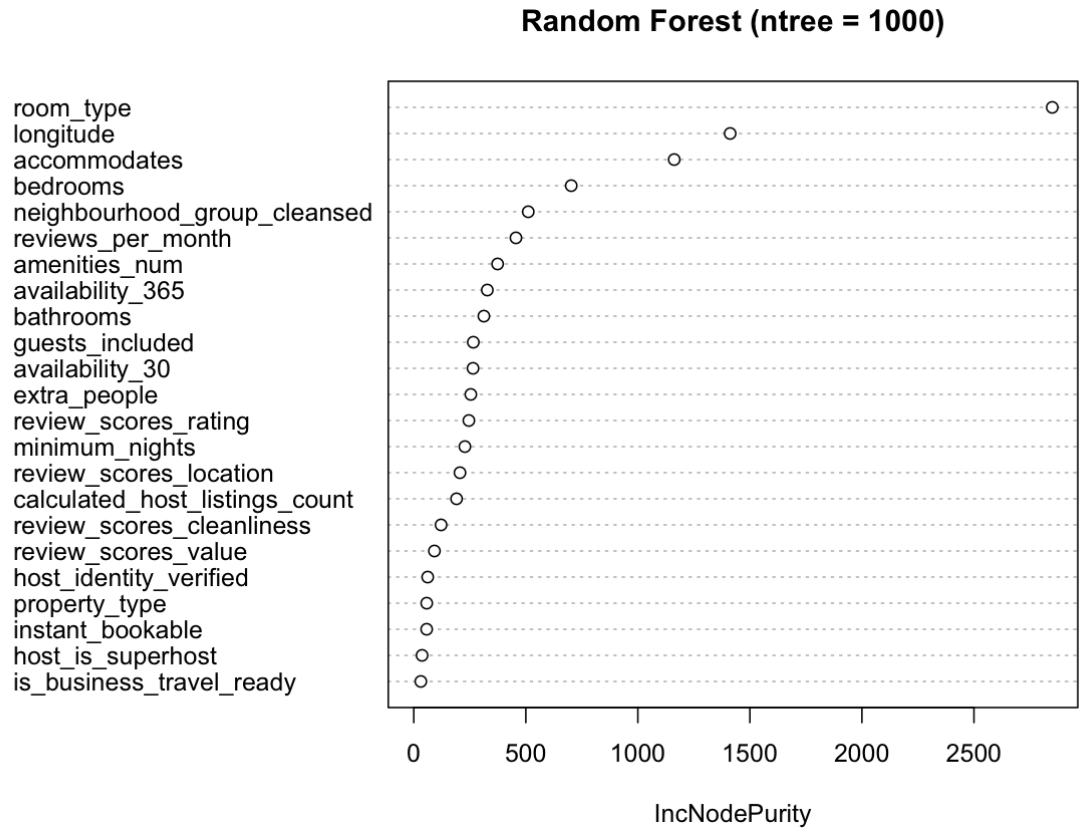
# 4. Predictive modeling

In the modeling stage, multiple algorithms were tested such as Multiple Linear Regression, Tree method with tuning, Random Forest, and Boosting Model. Other algorithms such as stepwise selection, XGboost, multiple linear regression with up to 2-way interactions were evaluated but their performances were not impressive. Thus, those four modeling methods became the finalists.

For the multiple linear regression method, model was fitted using the best 23 features above mentioned. As we do not have outcomes for test set, train set was used to calculate the RMSE and $R^2$ values: 0.415 and 0.622, respectively.

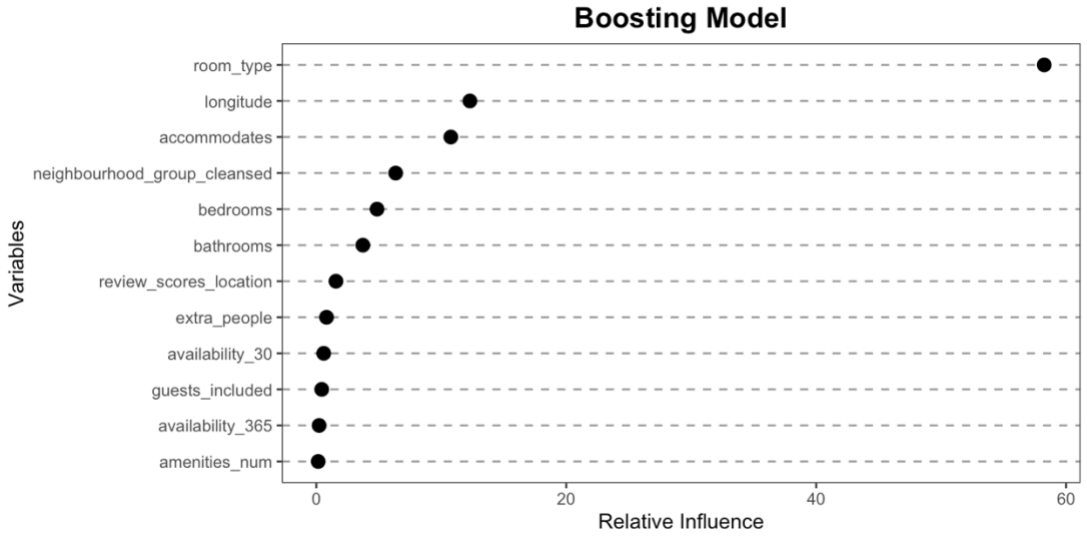
With the tree method with tuning, 5-fold cross-validation was also used together to determine the best complexity parameter with the lowest RMSE. The obtained RMSE with respect to the train set was 0.418 and the $R^2$ was 0.616.

For the Random Forest model, all the 23 features were input in the model. In accordance with [Figure 6] room type, longitude, accommodates were the top 3 most important variables (in order) among the selected features. The RMSE of the Random Forest model recorded 0.175 and the $R^2$ showed 0.942, which is the best performance.




[Figure 6] Dot chart of variable importance as measured by the Random Forest model

When it comes to the boosted regression model, [Figure 7] showed the similar results to the Random Forest. Likewise, room type, longitude, and accommodates remained the same as the top 3 most influential variables in the boosting model. The RMSE of the Random Forest model recorded 0.414 and the $R^2$ showed 0.633.




[Figure 7] Dot chart of variable importance as measured by the Boosting model

# 5. Model evaluation and selection

| Models | RMSE | R2 |
|---|---|---|
| Multiple Linear Regression | 0.415 | 0.622 |
| Decision Tree with Tuning | 0.418 | 0.616 |
| Random Forest | 0.175 | 0.942 |
| Boosting Model | 0.414 | 0.633 |

[Figure 8] Performance measured by RMSE across the models

Overall, Random Forest showed the best performance to predict the price with the lowest RMSE among multiple machine learning methods. Therefore, the Random Forest model was selected as our final model and submitted to the Kaggle competition.

One of the interesting findings is that price was highly associated with the longitude of the listing. We could infer that the NYC, especially Manhattan, is vertically longer rectangular shape. As most of tourist attractions are below Central Park, thereby around or below mid-town areas are the most preferred by Airbnb users, which increased the rental price higher. It appears as though as the longitude goes down, the price increases. Therefore, we could observe the significant negative correlation between the rental price and the longitude.

# 6. Failure and Improvements

I got through many failures when preprocessing the data, fitting the models, and submitting the output. In particular, converting amenities variable into meaningful information was difficult. I only counted the number of amenities and used it as one of predictors. I could have generated multiple independent variables of amenity types as dummy variables. For example, if one listing has a cable TV, then I should have generated a dummy variable with `cableTV_YN` that informs us whether the listing has a cable TV or not.

Moreover, data imputation would be one of the options to try. As we have many missing values in the data points. I chose to remove the variables with >20% NA values to minimize the bias in prediction. It would be better if I would have checked whether there are any specific patterns for missing values, and imputed with proper values.

One of the mistakes I made when generating a submission .csv file, I forgot exponentiating the result as I used the log-transformed price instead of the original price. I was pretty surprised about the high RMSE, then realized that I did not exponentiate the predicted price. It was easily corrected in the code afterwards.

Overall, it would be better if we have the Airbnb data over the course of years, as we would be able to analyze the pattern over time with respect to the rental prices.