**Kaggle Report**

**1. Introduction**

Kaggle describes this competition as follows:

This is a listing of over 25,000 Airbnb rentals in New York City. The goal of this competition is to predict the price for a rental using over 90 variables on the property, host, and past reviews, and generate a prediction for each id in scoringData.csv.

Note: the red text are the failed steps and the underlined text are variables.

**2. Exploring the Data**

**2.1 Load and examine the data**

First, I imported the analysisData.csv as train data and scoringData.csv as test data into R, and then examined the structure as well as the summary of train data. I observed the summary of train dataset and checked the number of levels of all factors in train to find out unnecessary variables. 32 variables of them are text description of the house, and some of them only contain NAs or only one level, so I deleted them from train and test dataset.

Next, I checked the structure of test dataset and found that “zipcode” is integer in test while it is a factor in train. I then convert “zipcode” in test into factor to keep the variable types consistent in train and test data. Then I combine test and train into one dataset named “all” to get prepared for further data exploration.

**2.2 Exploring some important variables**

**2.2.1 Explore the response variable: price**

First, I used qplot (Figure 2.2.1) to figure out the distribution of price. The distribution of price is right skewed which means the prices of most rentals range from $0 to $250. Therefore, a log(x+1) transformation of price should be conducted here.

Note: I used log(price+1) in the final modeling part at the beginning, but found the RMSEs of prediction are higher than those without log(x+1) transformation. Thus, the following steps are those without log transformation of price.

**2.2.2 Check and impute missing data**

Note: I failed to explore the correlations of variables with price at first because there are many missing values in some numeric variables. Thus, I decided to impute the missing data first.

First of all, I checked the completeness of the data and find the following variables with missing values and the amount of NAs.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| monthly\_price | square\_feet | weekly\_price | security\_deposit | cleaning\_fee | zipcode | beds | reviews\_per\_month |
| 36134 | 35977 | 31644 | 14715 | 7104 | 93 | 19 | 1 |

I handled each variable separately here. For categorical feature zipcode and numeric feature security\_deposit, it is reasonable to replace NA values with the most frequently occurring value for that variable. For other numeric variables such as reviews\_per\_month, cleaning\_fee, weekly\_price, monthly\_price and square\_feet, I replaced NAs with their medians. Finally, I replaced NAs in beds with 0 because we can’t assume there is a bed in the rental without any information provided.

**2.2.3 Correlations with price**

This section aims to find the important numeric variables. I used corrplot package to draw a correlation plot to visualize the correlations of 32 numeric variables with price.

According to the correlation plot (Figure 2.2.3), there are two predictors with relatively higher correlation with price: accommodates (0.58) and cleaning\_fee (0.59).

Also, multicollinearity is an issue. For example, availability\_30, availability\_60, availability\_365 and availability\_90 are highly correlated with each other, and have similar low correlations with price.

**3. Preparing the Data for Analysis**

**3.1 Drop highly correlated variables**

In this section, I used the correlation matrix in the previous section to inspect if there are two variables are highly correlated. And between them, I dropped the variable with the lower correlation with price.

For example, availability should be positively correlated with price. However, according to the plot, it is negatively correlated with price, so I dropped availability\_90 first. Between availability\_30 and availability\_60, I kept availability\_30 as its correlation with price is 0.01, larger than that of availability\_60.

In the same way, when choosing variable between review\_scores\_value and review\_scores\_rating, I compared their correlations with price, and kept review\_scores\_rating. Then I dropped review\_scores\_communication, review\_scores\_accuracy and review\_scores\_checkin and kept review\_scores\_cleaningness. I kept some highly correlated variables such as beds and accommodates because their correlations with price are relatively high.

**3.2 Drop categorical variables with more than 53 levels (only for random forest)**

Note: I tried random forest here and found error message said some variables have number of levels larger than 53 which prevent random forest from working.

I checked the number of levels of factors, and deleted the variables with more than 53 levels.

**3.3 Select features for modeling**

First, I carried out a Hybrid Stepwise Selection and got 31 selected variables.

Next, I wanted to get an overview of the most important variables including the categorical features. I applied the variables selected by the Hybrid Stepwise Selection into the Random Forest Model. Then I checked the IncNodePurity (Figure 3.3) and dropped the least important variables: host\_has\_profile\_pic and square\_feet.

**4. Modeling Techniques**

**4.1 Linear Regression**

The first model I tried was linear regression without feature selection described in 3.3. The result was not satisfactory. Next, I incorporated Hybrid Stepwise feature selection and used the variables selected to build a linear regression model. The results turned out to be much better, but still Higher than 60.

**4.2 Random Forest**

After trying several linear regression models, I turned to work with Random Forest.

Initially, I applied log(price+1) transformation and got RMSEs around 55 on leaderboard. Then I tried the models without log transformation and found the results were improved. The best prediction I got was obtained by following the steps described in the previous four sections.

**5. Results**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| RMSE on Kaggle | 66.80820 | 62.42154 | 60.19873 | 55.88143 | 55.49850 | 53.56626 | 53.54182  (Best) |
| Model | Linear Regression | Linear Regression  (after further imputing missing data) | Linear Regression (after feature selection) | Random Forest  (with log transformation) | Random Forest  (with log transformation; after Hybrid Stepwise selection) | Random Forest  (without log transformation; after Hybrid Stepwise selection) | Random Forest  (without log term; after selecting feature as described in section 3.3) |

**6. Appendices**

**6.1 R Markdown**

Please check the attached word file “kaggle\_report\_code.docx”.

**6.2 Figures**

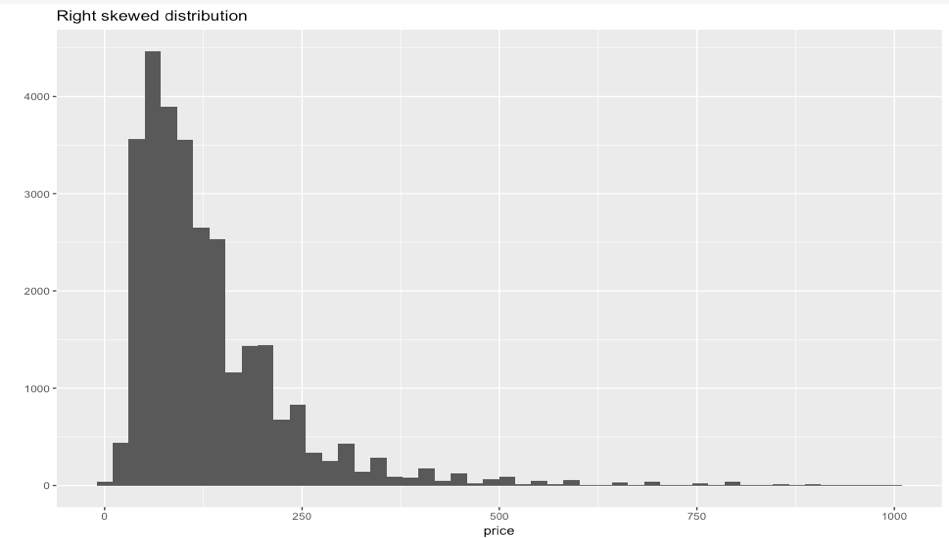
****

Figure 2.2.1 Right skewed distribution of price

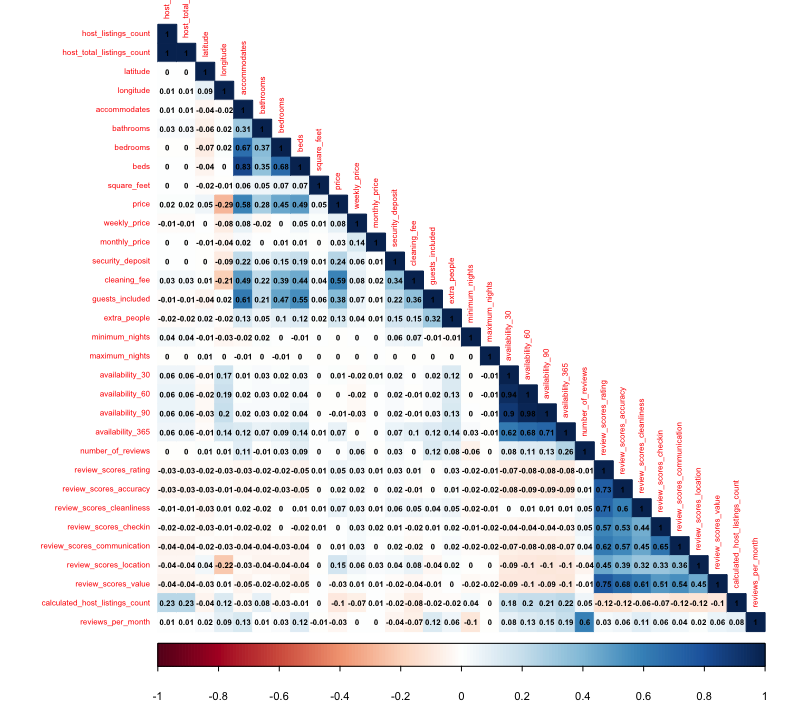
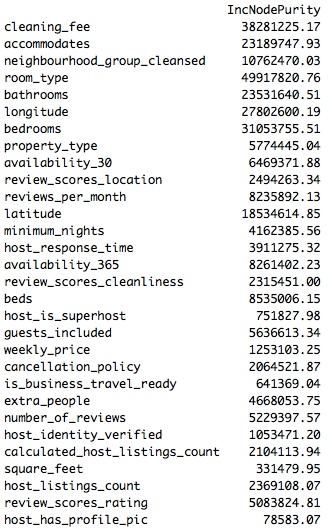
****

Figure 2.2.3 Correlations with price

Figure 3.3 Variable importance by Random Forest