A Survey on Machine Learning Techniques for Airbnb Price Prediction

STAT 493: Senior Seminar Project

Submitted By:
Kevil Khadka

Prof. Darrin Weber
Department of Mathematics and SDS
University of Evansville

Abstract

In this report, I develop a model to predict the Airbnb rental Price of Los Angeles, California (USA) based on their listing dataset, and also, I work on text analysis and sentiment analysis based on Airbnb's review dataset. This model is very helpful to find a price difference based on different characteristics like room type, minimum nights staying, neighborhood area, etc. Plus, it helps tourists to find out the best destination and right place to stay based on a review analysis of host listing. The dataset I used was provided by Airbnb's website called Inside Airbnb. The problem is modeled as a classification problem and found random forest method and naïve Bayes to be a very best model to predict the price based on the accuracy while multiple linear regression has the smallest Test MSE comparing to other models. The purpose of the project is also to find the best models based on accuracy and test mean square error (MSE). I use the following methods: Multiple Linear Regression, Decision Tree, Naive Bayes, Random Forest, Bagging, Linear Discriminant Analysis, Quadratic Discriminant Analysis, and k-Nearest Neighbor. For the text analysis and sentiment analysis, I use the review datasets for the following cities: Los Angeles, Chicago, New York, Boston, London, and Greater Manchester. I use the concept of Term frequency-inverse document frequency (*tf* - *idf*) to find out how important a word or comment is to a document or in a collection of documents, which focuses the difference between the reviews of the two cities.

1. Introduction

Airbnb is a vastly growing and a committed online marketplace that unites people who have house properties and tourists who are attracted to rent short-term or long-term lodging. With the growth of internet-facilities, Airbnb is gaining large support to the peer-to-peer economy growth (Wu 1). Airbnb site allows consumers a variety of services concerning rooms, locations, adventures, prices range to choose from. Since the company started in 2008, it has now over 150 million total users, more than 2 million people staying in an Airbnb per night, 6 million global Airbnb listings worldwide, and 35-billion-dollar valuation based on recent stock sale (Airbnb-statistics). The global compound growth rate of Airbnb is increased to "153%" since 2009. Looking at this significant development, Airbnb's competitors would easily replicate the same business model which makes long-term growth further challenging. As we see very similar structures (low fees, easy and fast accessible service) within the industries, there is limited differentiation found among the companies' core business models which are the number and diversity of listings, performance on the platform, customer, and worker relationship. This project focuses on those differentiations and how machine learning algorithms could be helpful to predict a better model.

Unlike hotels, Airbnb prices are usually determined by the hosts. For new hosts and existing hosts with new listings, it is rather complicated to determine the prices of the listings without losing their popularity. For the consumers' side, it is challenging to find out the perfect host based on the listed price, neighborhood area, reviews, etc. The minimum night price for Airbnb room depends on multiple factors, like locations, neighborhood area, room type, and I change many variables into categorical, continuous, text, and data features.

I divide the project into three parts; data visualizations, machine learning algorithms, and natural language processing (NLP). In the data visualization part, I focus on finding the impact of predictor variables on the response variable (price). I analyze each variable and find the relationship between them through the approach of simple linear regression. Other works like analyzing missing values, data cleaning, and selection of suitable variables are done under this part one.

The machine learning algorithms come in part two, where I use different methods to develop the best model to predict the price. I plan to change the numeric response into a categorical variable which I use as a response in all models. I successfully try using eight methods of

machine learning algorithms. In the final part, I use the review dataset to do text mining and sentimental analysis. I wanted to find out the various aspects of the rental experience people like and dislike. In-text analysis, I use unigram, bigram, and trigram models to construct a word cloud and compare the cities based on *tf - idf* score. Similarly, in sentiment, I analyze the review dataset to find out the various positive and negative words people posted for their Airbnb hosts. I conclude the sentiment analysis by finding out the most frequent word associated with "anger" and "trust" using semi-join.

2. Dataset and Features

2.1. Dataset

The Airbnb already has the collection of listing and review datasets of popular cities. I use the listing dataset of Los Angeles, which was posted on 6th December 2018. I downloaded the review dataset of the other five cites of the same date. The Los Angeles dataset contains 42,992 observations. While the review datasets of each city contain more than millions of observations.

The listing dataset I used to study this project had sixteen variables where I remove some of the unimportant variables from the dataset like id, last reviews. The detail information about each variable are:

id: Listing ID
name: Listing Title
host_id: ID of Host
host_name: Name of Host
neighbourhood_group: Borough that contains listing
neighborhood: Name of neighborhoods that listing is in
latitude: latitude of listing
longitude: longitude of listing
room_type: Type of public space that is being offered
price: price per night, USD
minimum_nights: minimum number of nights required to book listing
number_of_reviews: total number of reviews that listing has accumulated
last_review: date in which listing was last rented
reviews_per_month: total number of reviews divided by the number of months the listing is active
calculated_host_listings_count: amount of listing per host
availability_365: number of days per year the listing is active

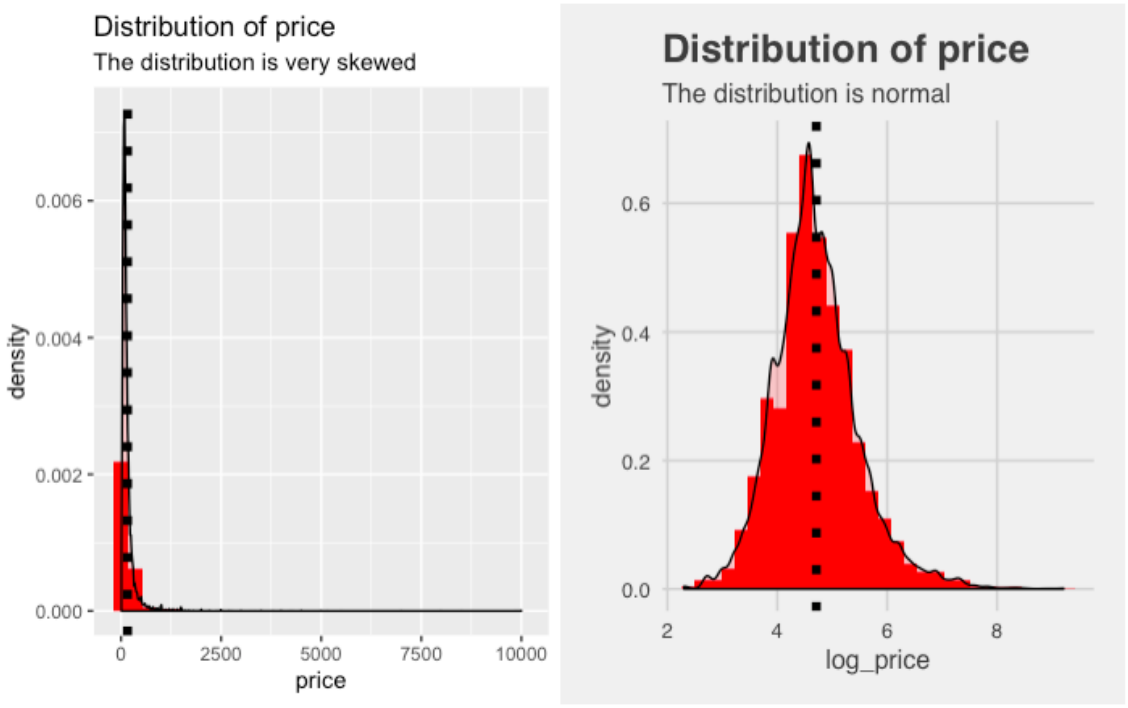
The review dataset is composed of six variables, and out of six, only two variables are used to do text analysis and sentiment analysis. Only in Los Angeles review dataset, there are 1,226,674 observations so I plan to use only 1000 observations. The two variables I use are: hosting_id and comments.

2.2. Features

On this section, I am going to talk about the most important variables which I use to predict the response variable. I eliminated the features like id, name, host_id, host_name, last_reviews which appear to be non-beneficial, and features like latitude, longitude, neighborhood are deleted after later analysis.

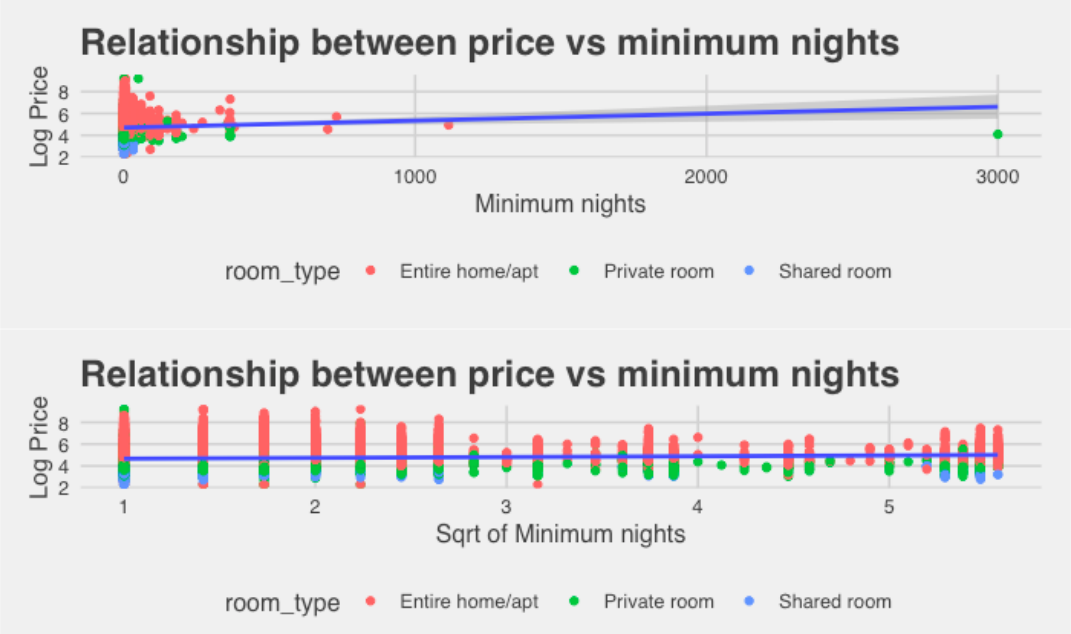
- Price (Response)

Price is the response variable for this project. I filter out the price value where the price is range from 10 dollars to 10000 dollars. As there exist abnormally high prices in the dataset, I have to use the logarithmic transformation of the price variable to get a normal distribution. For the machine learning part, I focus on splitting the price into three categorical levels: low, medium, and high. Each price level was decided by taking out a mean of three and divided by 3. The figure below shows the distribution of price before and after log transformation.



- Minimum_nights

It is a predictor variable in which values range from 1 to 31. I delete some observations which have more than 31 value. It is best to keep the minimum nights under 31 as it is equal to a one-month duration. The distribution of the variable is very skewed, so I use square root transformation to get a proper normal distribution of minimum nights variable. I rename the variable into "sqrt.min.nights".



- Availablity_365

This predictor is also used in the final model. It denotes by days the listing is active. There are no changes made to this variable except I keep the value range of 1 to 365.

Variables like number_of_reviews and reviews_per_month are changed into the categorical variables as price response. For the number_of_reviews variable, there are three levels based on the total reviews the listing has accumulated. Low level (LR) has review number less than 10, the Medium level(MR) has less than 70 and the High level (HR) has more than 70 reviews. I replace the variable "number_of_reviews" with "num.review.level". I use the same technique for "reviews_per_month" variable and replace the variable with "review.per. month".

3. Statistical Model

After cleaning and exploring the dataset and keeping the important variables, I split the dataset into train: test with a ratio of 70:30. The methods I use to find out the best predictor variables to predict the response variable are described below:

3.1. Forward and Backward Elimination Method

It is also known as the stepwise method which helps to select the best variables to use for the final model. The forward elimination method chooses a subset of the predictor variables for the final model. It starts with a null model with zero predictors, just one intercept. It picks the variable with the lowest residual sum of squares. Unlike the forward method, a backward elimination method begins with the full model containing all predictors, and then iteratively removes the least useful predictor, one at a time.

After applying the following method, the selected predictor variables for the final model are "neighbourhood_group", "room_type", "availability_365", "sqrt.min.nights", "num.review.level", "review.per.month" and "host.list.count". I use the BIC plot as well as the regsubset method (leaps package), the result was different. But I use the final solution stepwise method since it was a reasonable idea to keep the meaningful variable to predict the price.

3.2. Multiple Linear Regression (MLR)

It is known as a simple statistical technique that uses several explanatory variables to predict the outcome of price response. The goal is to model the linear relationship between the explanatory variables and the response variable.

The coefficient of determination (R-squared) is used to measure how much of the variation in outcome can be explained by the variation in the explanatory variables. The R-squared of the model from the project is 0.4477, which means 44.77% of the outcome can be explained by the model. The Residual standard error was 0.3330, where the p-value is below 0.05 that means the model is statistically significant.

I run the test model in multiple linear regression and find out the Test mean square error of 0.3387. Simply, "MSE, defined as the sum of the squared residuals divided by n – p (n = number of observations, p = number of coefficients), is an unbiased estimator for the error variance in a model" (R Package Documentation).

3.3. Decision Tree

It is a graph to represent the best results in the form of a tree. The nodes in a graph represent as best variables that explain the response variable. It uses the top-down, greedy approach which simply means the best and fast way to choose the variable. The R packages "*rpart*" and "*rpart.plot*" are used to create decision trees. After running the test dataset in the decision tree

model, the accuracy was found to be 0.6247 and the test MSE was 0.3752 which is quite larger than multiple linear regression.

3.4. Naïve Bayes

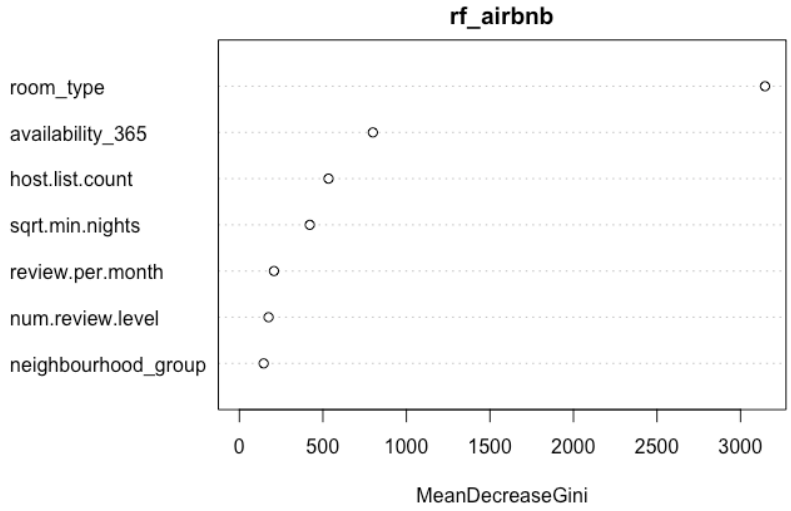
It is a supervised machine learning method. It calculates the conditional probability that is the probability of even occurring based on information about the events in the past.

After running the test dataset in the naïve Bayes model, the accuracy was found to be 0.6336 and the test MSE was 0.3663 which is quite larger than multiple linear regression but smaller than decision tree.

3.5. Random Forest

It is the best method so far to achieve lower test MSE and higher accuracy rate among other methods. The accuracy of test dataset was 0.6459, and the test MSE was 0.354089 which is still higher than multiple linear regression' test MSE.

The function *varImpPlot* () finds out the "room_type" as the important variable in the random forest.



3.6. Bagging

It is similar method as the random forest with m = p (m = number of predictors = p). I use square root of p to find the best model with the accuracy of 0.6312 and the test MSE of 0.3866.

3.7. Linear Discriminant Analysis (LDA)

LDA works for factor analysis to look for linear combinations of a variable to best explain the response. Since the response is a categorical type, LDA tries to explain the difference between the classes of data. From the result of prior probabilities, 33.92% of our observations are

low price estimates whereas 33.35% of High-level price estimates. Running the test data helps to find out the accuracy of 0.6273 and the test MSE of 0.3726 which is quite large.

3.8. Quadratic Discriminant Analysis (QDA)

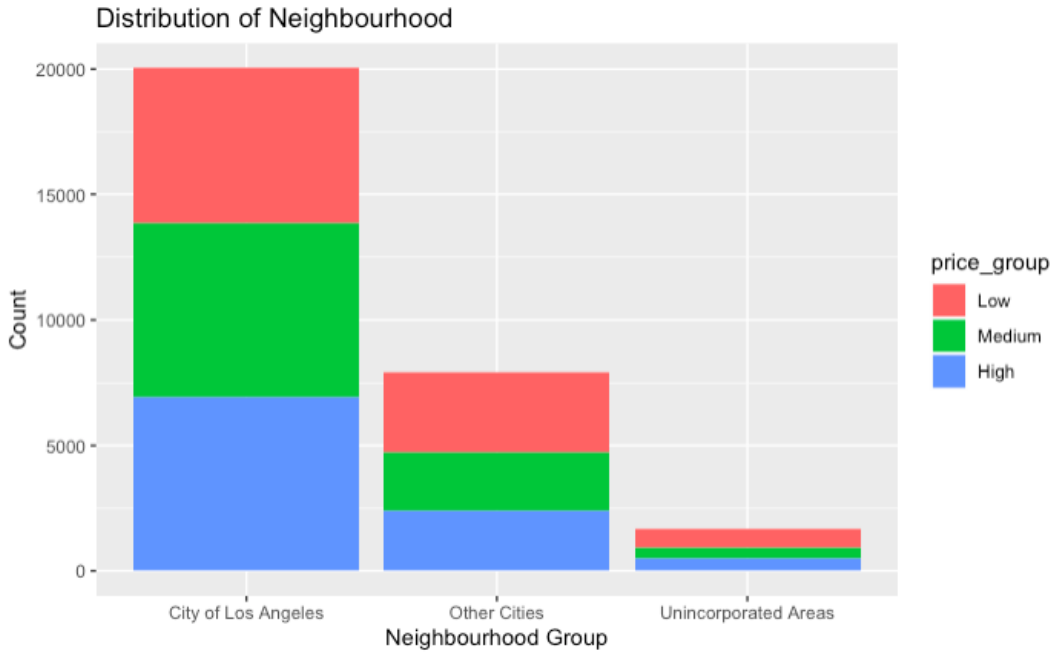
QDA works similar to LDA. It helps to introduce more flexibility but has more variance too. The result of prior probabilities is also similar to LDA. Running the test data helps to find out the small accuracy of 0.5159 and the test MSE of 0.4841 which is worst test MSE rate so far.

3.9. k-Nearest Neighbor (kNN)

For kNN method, I split the dataset into train: test with the ratio of 70:30. It works for both categorical and numerical response. For the model, I changed all predictors variable into numeric variable to find out the best distance. The test MSE for kNN model was 0.4617 and the accuracy was 0.5383. The performance of kNN method was not the best as expected.

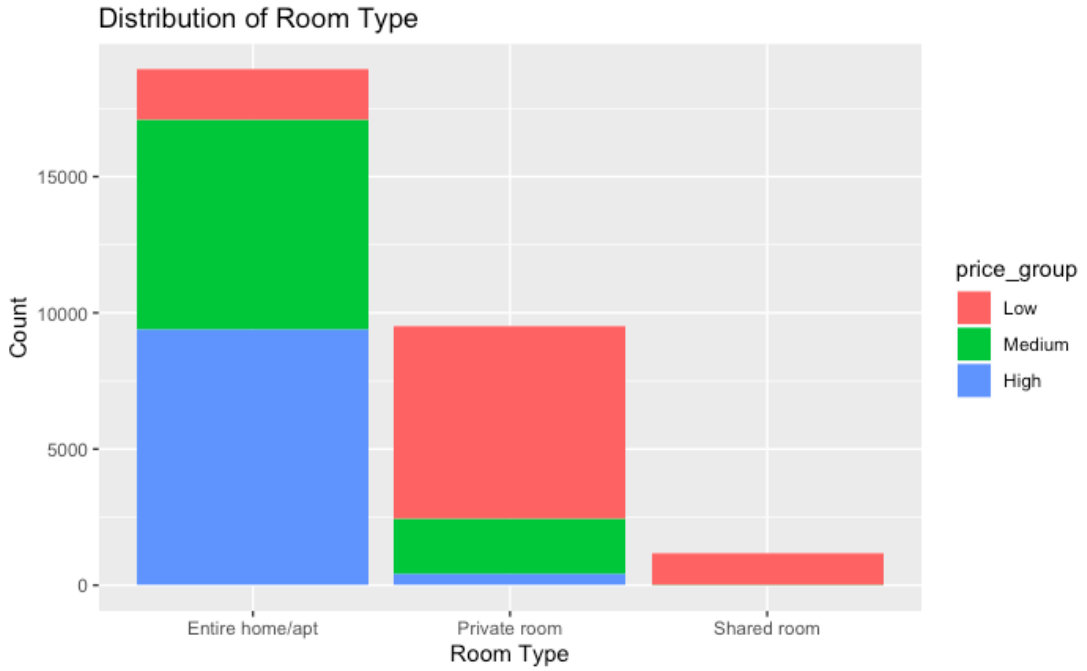
4. Analyze the result

For the data visualization part, a bar plot of the "neighbourhood_group" shows the City of Los Angeles as the major attraction place. Los Angeles city seems to be pretty expensive comparing with other. The below figure shows the distribution of neighbourhood:



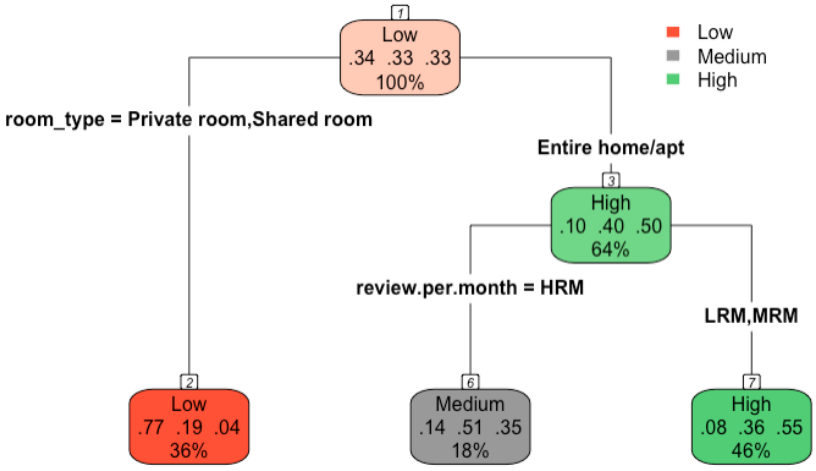
Another best predictor to predict the price is "room_type". Looking through the bar plot of room type, I conclude that the tourists are likely to book the entire home or apartment rather than

sharing the room with others. But booking the entire home comes expensive while the private and shared room has a cheaper rate.

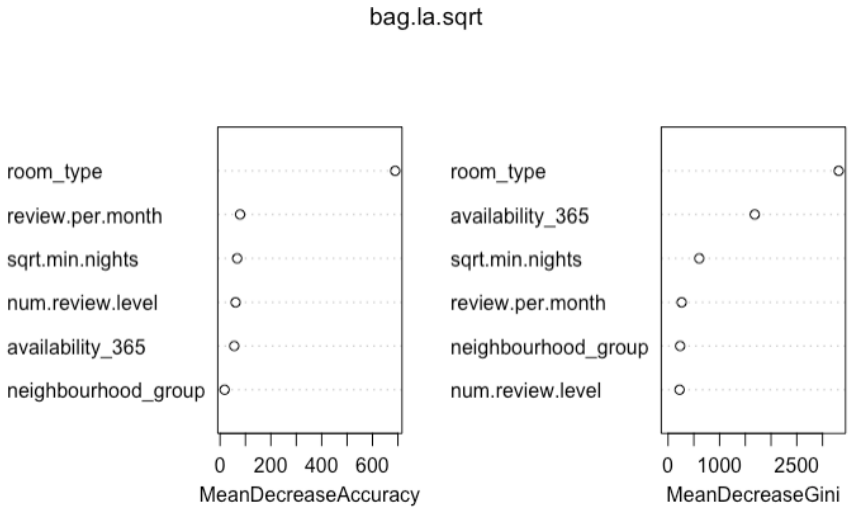


It was important to know how much people love to comment on their host and the place. With respect to the types of room, people are less likely to leave a comment when they stayed in shared room.

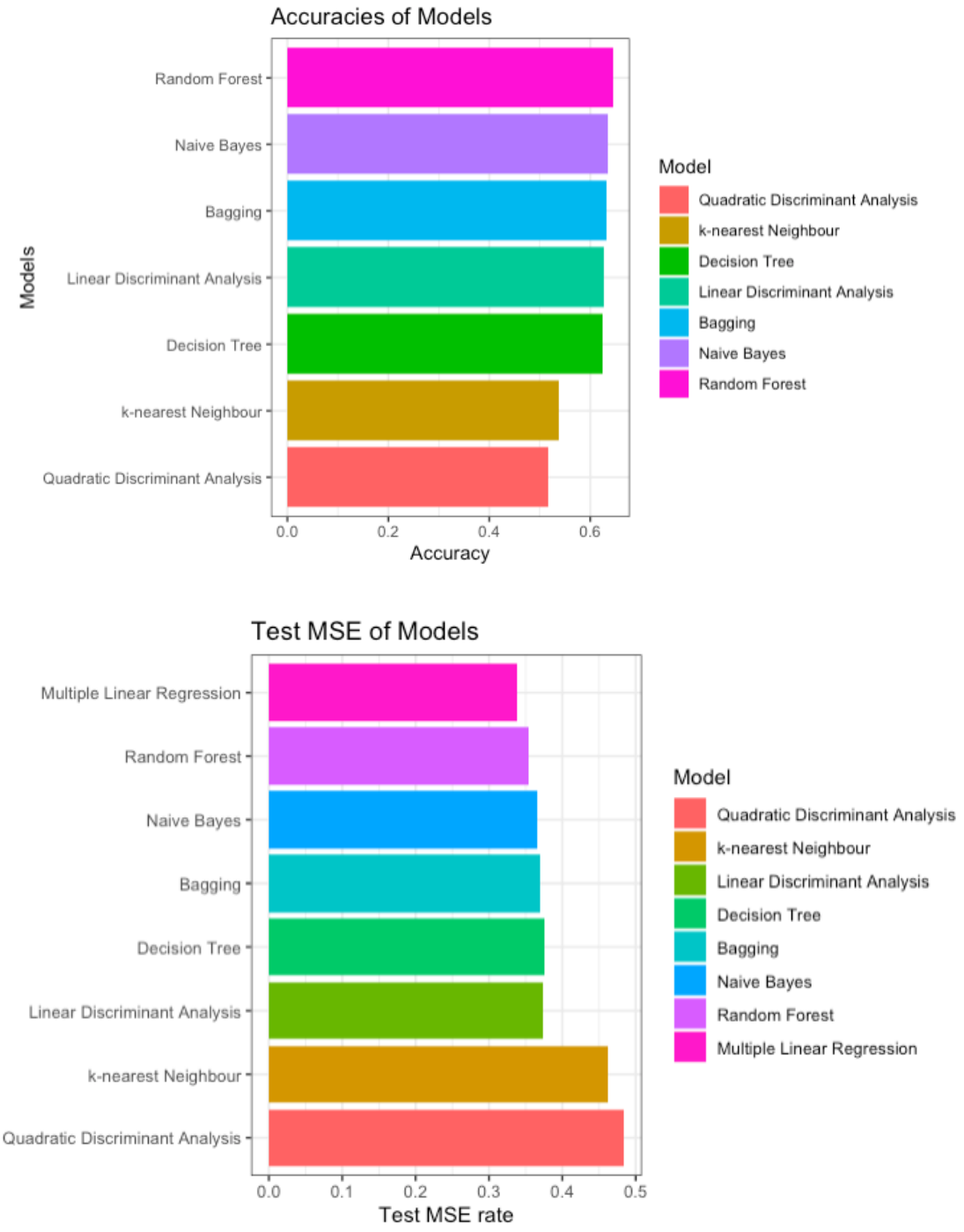
From Decision Tree method, I obtain the test accuracy of 0.6247 and test MSE of 0.3752. The tree uses 3 nodes where “room_type” seems important variable to predict the response. The figure below is the decision tree:



The test accuracy of random forest was 0.6459 and the test MSE of 0.354089. The OOB estimate of error rate was 35.8%. The plot shows “room_type” as the important variable. The plot from bagging method shows “room_type” and “availability_365” as important variables.



Until now, I found out the result of a problem regarding Airbnb attributes. In the machine learning part, I try to find out the best model with the least test MSE rate and higher accuracy rate. After trying all suitable methods, the random forest, Naïve Bayes and Bagging seem the best choice to predict the price response. While multiple linear regression had the smallest test MSE rate comparing to other models.

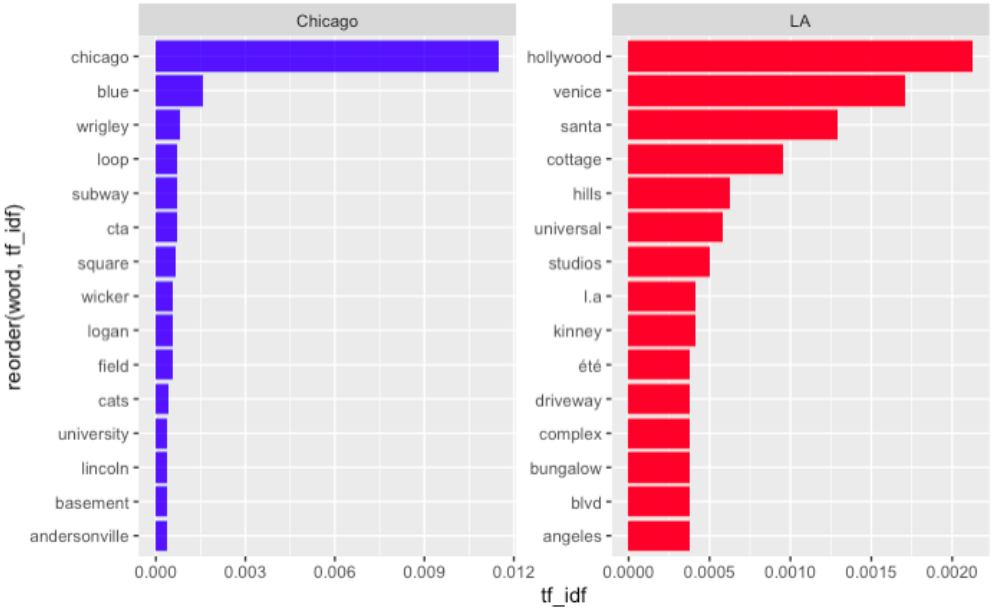


In part 3, I did natural language processing (NLP) of the review dataset. One is text mining and the other is sentiment analysis. We looked into the datasets of six major cities: Los Angeles, Chicago, New York, Boston, London, and Greater Manchester.

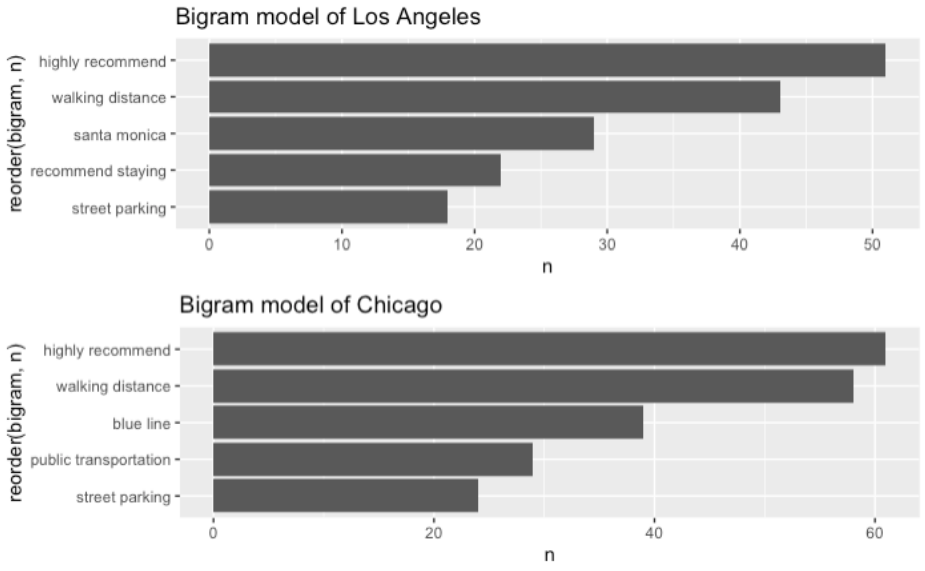
For text mining, I generate the word cloud to look into similar comment people posted on Airbnb. It seems the word cloud for each city shares most of the similar words. I develop the three models to work on text analysis which are: unigram, bigram, and trigram model.

Unigram model selects the most eye-catching single word from the review dataset. As I mentioned earlier, each review dataset has 1000 observations. Looking at the below word cloud of Los Angeles and Chicago, we find common words like clean, nice, comfortable, etc. Similar comment goes for other cities too.

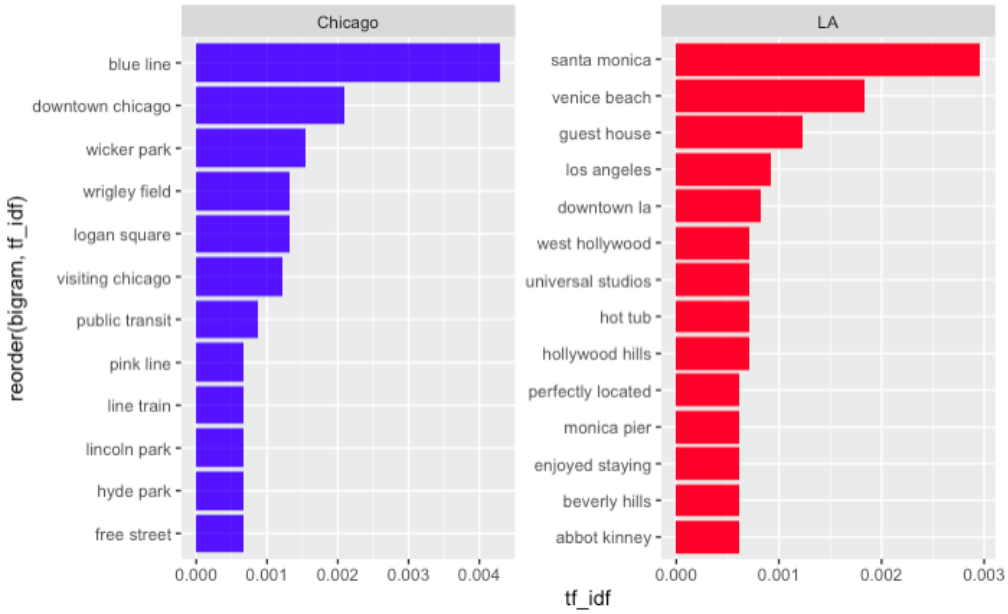
The Term frequency-inverse document frequency (*tf-idf*) of Los Angeles and Chicago shows that “hollywood” and “chicago” terms are most popular in the review dataset. “Venice” and “santa” seem important terms in the Los Angeles dataset which are the main attraction places in the city. I have generated the *tf-idf* score plot for New York and Boston and London and Greater Manchester. It is in the appendix part.



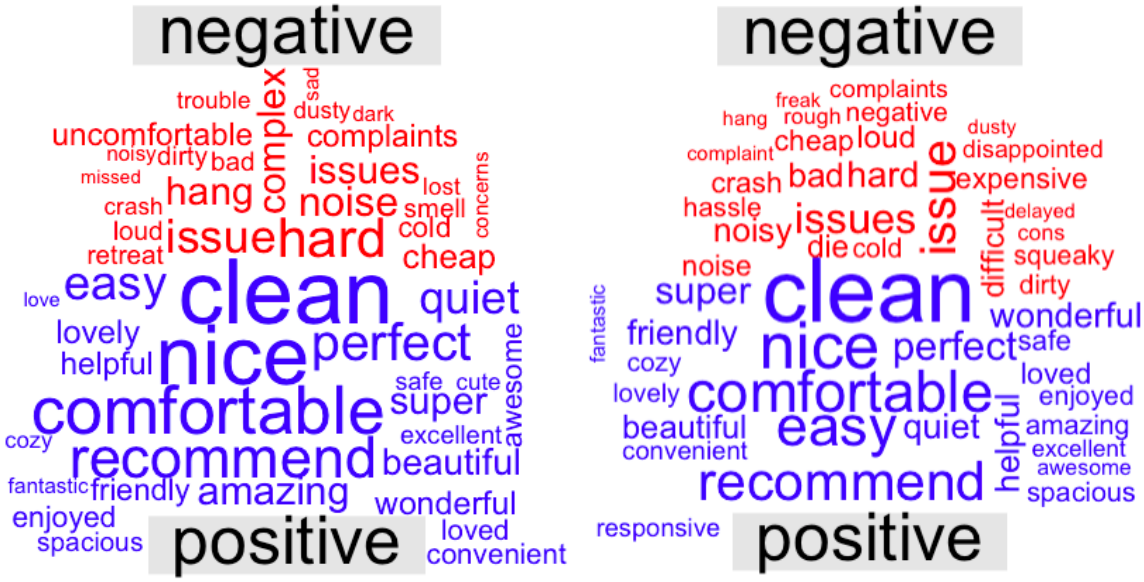
The bigram model which simply means the combination of two words. We can see the bigram model choose two most common word use in a review dataset. From below figure, “highly recommend” and “walking distance” are mostly used words by the people in Los Angeles and Chicago city.



And the *tf-idf* of bigram model of LA shows “santa monica” is mostly important place whereas in Chicago, “blue line” seems most common word. In appendix part, there are models for other cities too.



In the sentiment analysis part, we start by explaining what sentiment analysis is. According to tidytextmining, "Sentiment analysis is one of the most obvious things we can do with unlabeled text data (with no score or no rating) to extract some insights out of it. One of the most primitive sentiment analyses is to perform a simple dictionary lookup and calculate a final composite score based on the number of occurrences of positive and negative words" (“Text Mining with R”). I analyze the sentiment part using two common lexicons, bing and NRC. Using the bing lexicon, I find out the most positive and negative words in the review dataset of all cities. To get an idea of how it looks like, I post the word cloud sentiment done in the Los Angeles dataset. The below first picture the sentiment analysis of LA and the second one is of Chicago. The most positive words found in both cities are clean, nice whereas the negative words are an issue, hard, etc.



"The nrc lexicon categorizes words in a binary fashion into categories of positive, negative, anger, anticipation, disgust, fear, joy, sadness, surprise, and trust" ("Text Mining with R"). I look into the common positive word "trust" which seems same in all cities. We find the score of trust from the NRC lexicon and see the similar common word associated with trust in each city dataset. The result shows that the most positive or good words are about the room, neighborhoods area and host.

Similar way, we research the common negative word "anger", and the result shows that the most negative words are about the expensive room, noisy neighborhoods area, or sometime about the room when there is no AC/heater.

5. Future Work

First of all, the size of the dataset was very huge which requires the best computer processor to run the data. I tried to use cloud computing but did not succeed. So, I am planning to study cloud computing which is helpful to run and analyze big data in a fast way.

The results are not as good as expected. The accuracy rates of all models are below 0.70 and the test MSE rates are above 0.40. I think the assumptions or changes I made on variables were the wrong ideas. In the future, I will try to work on the same methods using depth analysis on the dataset. My planning is further remodified kNN method since it performed very bad comparing to other models otherwise, I will choose to replace it with Radial Basi Functions (RBF). Also, I will add some new machine learning techniques like GAM (Generalized Additive Model),

Neural Network to predict the price. Due to the short time, I could not able to compare the hotel industry and Airbnb using review dataset. These are the work I would like to complete in the coming days.

6. Conclusion

Through the classification and sentiment analysis, I able to come up with the answer that the price of Airbnb in different cities is mostly determined by their room-type and availability throughout the year. In machine learning algorithms, random forest and bagging perform the best than the rest models. The accuracy of all models except QDA and kNN are above 60% only, which suggests that there might be an absence of some potential factors I did not use in the model. The text analysis and sentiment analysis are a good way to understand consumer choice. The reviews are similar across the different cities. It seems many tourists leave positive reviews and use similar positive words to describe their rental experience. It also shows the friendly aspect of the Airbnb community. The *tf-idf* score suggests the popular tourist destinations and ways of transportation of each city. It might be difficult to conclude which city has a higher Airbnb rating since most of the reviews are positive and similar.

Bibliography

Silge, Julia, and David Robinson. “Text Mining with R.” 2 Sentiment Analysis with Tidy Data, Mar. 2020, www.tidytextmining.com/sentiment.html.

Wu, Steve, Lee, Frank, and Jeremy Reynald. “Airbnb”, *UCLA*, print, pp.1-13.

“Airbnb Statistics [2020]: User & Market Growth Data.” IPropertyManagement.com, IPropertyManagement, 2020, ipropertymanagement.com/research/airbnb-statistics.

“Inside Airbnb Adding Data to the Debate.” *Get the Data – Inside Airbnb*, Airbnb, 2020, insideairbnb.com/get-the-data.html.

“R Package Documentation.” R Package Documentation, rdrr.io/.

Appendix

R file:

Final_Coding_Khadka.rmd

- The following file contains r code of this project. The code analyzes the listing dataset and review dataset. The file itself is not big. Due to large amount of datafiles and codes, the file takes a longer time to knit.