Report

Wenyu Ji

November 30th, 2021

For this project, we are given a listing of 41,330 Airbnb rentals in New York City. This competition aims to predict the price for a rental using 90 variables on the property, host, and past reviews. I followed the procedure of CRISP-DM, which stands for Cross-Industry Standard Process for Data Mining, to guide me from exploring the data, preparing the data, and analyzing the data. In this project, I try to identify the attributes that affect the rental pricing of Airbnb, and I carried out the project in the following steps:

- Understanding Business Problem
- Data Cleaning and Preparation
- Data Visualization
- Modeling and Model Checking
- Evaluating and Finalizing the Model
- Prediction for the Final Result.

According to this analysis, I learned how to solve the real-world problem by applying predictive techniques in the course to a real-life example. I understand that different models have different characteristics. Some are effective while predicting the rental price, and some are not. I tried a couple of models and finalized the one that works the best, which is Random Forest. However, I believe there are other more accurate models that could predict the data. Therefore, I listed out some of my failed steps and missteps that I wish I could look deeper into and explore more to understand and clean the data. I will discuss each of the sections according to what I did, what I learned, my failed steps, and what I could do better.

According to the Understanding Business Problem, my goal is to accurately predict the rental price of Airbnb with the lower RMSE (root mean squared error) being the better. By doing the research online and personal experience, I originally, based on my common sense, got rid of some of the unimportant variables such as, id, name, summary, name, space, description, neighborhood_overview, notes, transit, access, interaction, house_rules, host_name, host_since, host_has_profile_pic, state, market, country, country_code, etc. (details refer to the code and Appendix A). These variables are not the critical attributes on determining the rental price. This step helped me save a lot of time before starting the data cleaning. However, eliminating too

many variables by self-consciousness is not always accurate. At first, I eliminated around 40 variables without using any feature selection; some key attributes I missed, such as minimum_nights_avg_ntm, maximum_minimum_nights, availability_30, calculated_host_listings_count_entire_homes, calculated_host_listings_count_private_rooms, etc. because some of the variables are very similar. I accidentally deleted the same attributes and missed significant value. Next time, I will carefully remove small amount variables and use feature selection such as Lasso Regression, Forward, or Backward selection to find the significant attributes.

According to Data Cleaning and Preparation, this is the lengthiest task, and I imputed and removed the missing value. I understand that effectively cleaning the data is the key to success. If I went back and did the project again, I would spend more time on data cleaning and try different ways to impute the missing value. To clean the data, I first removed the variables with more than 70% of N/As. By doing this, I got rid of weekly_price, monthly_price, and square_feet. After that, I used the median to replace missing values of these variables, such as cleaning_fee, beds, security_deposit, host_total_listings_count, host_listings_count, reviews_per_month. In the future, I would like to use a mean or 0 to replace the missing value. For example, some houses do not have a cleaning fee; in this case, if I put 0 instead of the median will be more accurate since manipulating the missing value will directly affect the accuracy of the prediction.

According to the Data Visualization, I first selected the numeric data and generated a correlation heat map demonstrating the relationship between the numeric variables (see Appendix B). This could help me quickly pick up on the significant variables. On the other hand, I should look deeper into the categorical data charts, such as a histogram chart listing out mean price comparison with all room types, mean price comparison between neighbourhood_cleansed and other categorical variables.

According to the modeling part, I split the dataset 70% for training and 30% for testing, and I ran Random Forest two times. The first time I used 50 trees to determine the important attribute by evaluating %IncMSE (percent increase in mean squared error). The second time I used 1000

trees for the final prediction (see Appendix C). I also tried linear regression, XGBoots, forest with Ranger and cross-validation which did not work the best. The reason why it did not work out is probably due to the way I cleaned the data. There are a number of predictive techniques we learned in the class; I found Random Forest produced the best result. However, many other techniques would be adequate to predict the price. If I did the project again, I would like to spend more time on data preparation stages to try different variables.

According to the final result, I got RMSE 43.79725 in R, and public score on Kaggle is 69.42407, and private score on Kaggle is 63.28060. By choosing the attributes, I understand that not all variables are predictive, and if I use too many predictors and attributes the model often overfits the data. At the end, I chose 45 important attributes as my final prediction variables, which is a large number of predictors. However, this gave me the smallest RMSE result. I understand that use too many predictors critically affected my results and led to overfitting. Therefore, it is essential to carefully select attributes and the number of variables used for the final model.

Finally, I learned the process of data exploration, summarization, preparation, and analysis. In the future, I would like to spend more time on the data preparation and cleaning stage to impute mean or 0 for the missing value and use different feature selection tools to test the important variables. I like how I calculated %IncMSE (percent increase in mean squared error) and listed them in order of importance. For numerical data, I like how the correlation heat map showed the correlation between each variable. For the categorical data, I would like to generate histograms to compare the difference between each category. Moreover, I could break down categorical data into dummy variables to better analyze the data. Predictive modeling is a repetitive exercise. I would like to apply what I learned to solve more the real-world problems.

Appendix A

Data Cleaning and Preparation

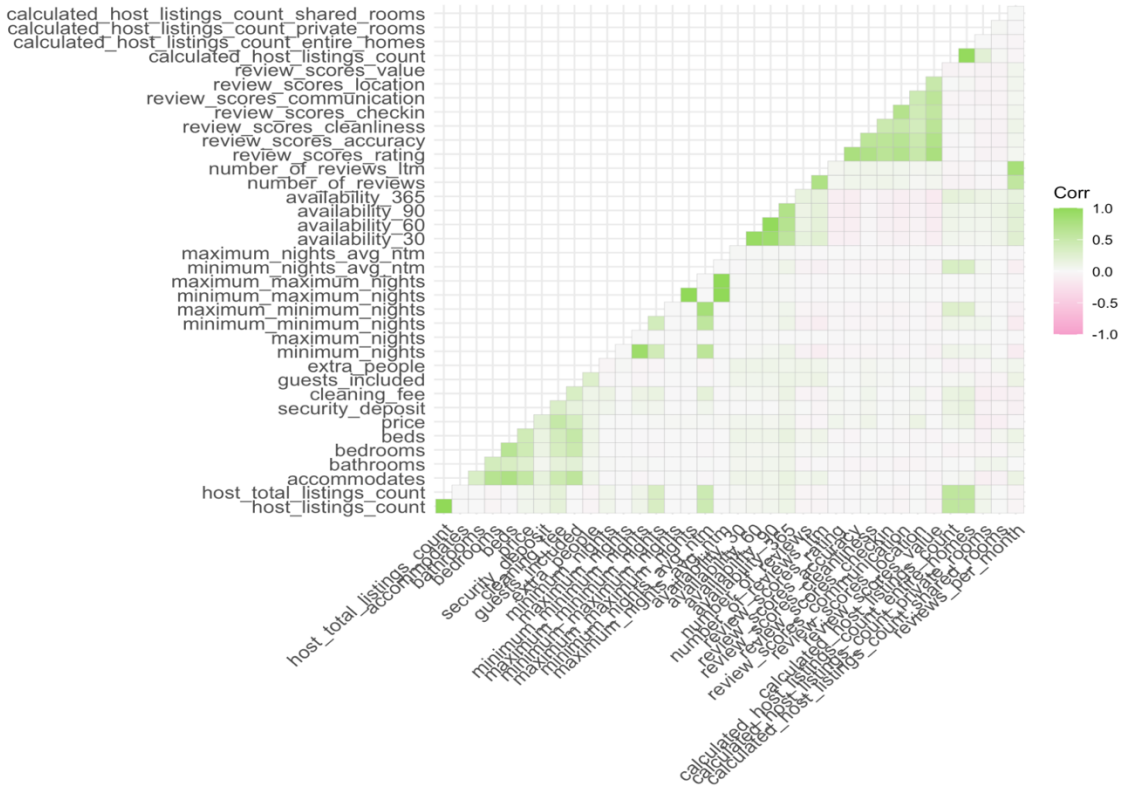
```r
data = read.csv('analysisData.csv',stringsAsFactors = F)

#subdata.
data.sub <- select(data, -id,-name,-summary,-name,-space,-description,-neighborhood_overview,-notes,-transit,-access,-interaction,-house_rules,-host_name,-host_since,-host_has_profile_pic,-state,-market,-country,-country_code,-square_feet,-weekly_price,-monthly_price,-requires_license,-license,-jurisdiction_names,-require_guest_profile_picture,-require_guest_phone_verification,-host_location,-host_about,host_verifications,host_verifications,smart_location,-host_is_superhost,-is_business_travel_ready,host_verifications,host_identity_verified,-host_verifications,-host_identity_verified,-first_review,-last_review,-host_neighbourhood,-host_acceptance_rate,-first_review,-last_review,-has_availability,-bed_type,-is_location_exact,-smart_location,extra_people,-city,-street,-cancellation_policy)
sort(sapply(data.sub, function(x) { sum(is.na(x))} ))
```

```r
#Fill remaining NAs with medians.
data.sub$cleaning_fee[which(is.na(data.sub$cleaning_fee))] =
median(data.sub$cleaning_fee,na.rm=T)
data.sub$beds[which(is.na(data.sub$beds))] = median(data.sub$beds,na.rm=T)
data.sub$security_deposit[which(is.na(data.sub$security_deposit))] =
median(data.sub$security_deposit,na.rm=T)
data.sub$host_total_listings_count[which(is.na(data.sub$host_total_listings_count))] =
median(data.sub$host_total_listings_count,na.rm=T)
data.sub$host_listings_count[which(is.na(data.sub$host_listings_count))] =
median(data.sub$host_listings_count,na.rm=T)
data.sub$reviews_per_month[which(is.na(data.sub$reviews_per_month))] =
median(data.sub$reviews_per_month,na.rm=T)
summary(data.sub)
```

Appendix B

Data Visualization (Correlation Heat Map)



Appendix C

Random Forest Code

```r
#Run randomForest determin importance features
library(randomForest)
set.seed(1031)
rf<- randomForest(price~.,data=data.sub,importance=TRUE,type='regression',ntree=50)

#select features

importance_rf <- as.data.frame(rf$importance)
importance_rf$`%IncMSE`
features=rownames(importance_rf)[-length(importance_rf)]
data_selected=data.sub[,c(features,'price')]
summary(data_selected)
```

```r
# make predication
library(randomForest)
set.seed(1031)
rf_selected <- randomForest(price~.,data=data_selected,importance=TRUE,type='regression',ntree=10
00)
pred = predict(rf_selected,newdata=test.sub)
rmse_forest = sqrt(mean((pred-test.sub$price)^2)); rmse_forest

```

```
[1] 43.79725
```