Executive Summary

Our team was invited to do a market data analysis for Airbnb. There are three main materials provided by Airbnb: a traveling dataset (csv file) that includes the average rating of the homestays, a test dataset (csv file) that does not include the average rating of the homestays, and an excel file that introduces the various indicators in detail. After screening and clearing the data, our team determined to focus on two main goals.

The first goal is to use the known information of the homestays that have been registered in the Airbnb brand to predict the new collaborators that will be selected in the future. After this process, we will get a prediction of the new collaborators' average ratings before they enter the market. The second goal is to understand the potential connections between variables by analyzing known data, and to locate the factors that are most likely to affect the average rating in the process of improving service quality in the future. In the prediction, we deleted two variables, for example "overview" which has a strong personality and, "country" which is the variable with most data convergence.

We also dealt with the missing values in the data set by filling or transforming them according to the definition of variables. When a value, for instance "monthly_price" is missing, we think that the merchant does not have a discount for monthly rent, so we convert this variable to the discount. For missing values such as "security_deposit" and "cleaning_fee", we assume there is no cleaning fee and no deposit.

Our main analysis method is modeling analysis, including but not limited to "Excel" and "RStudio". After trying Linear Regression\ KNN \ Random Forest \ Lasso \ Ridge and other models, we believe that the prediction result of Random Forest is the most accurate. There will be an error between the predicted value and the actual value. The larger the error, the larger the RMSE, and the smaller the error, the smaller the RMSE. Therefore, we judged the prediction accuracy of the model according to the predicted RMSE value. Among all the tried models, RMSE of the random forest is around 7.01, so we choose this model to make accurate predictions. We mainly judged the relationship between each index and the predicted value, based on the conclusion of linear regression. This model has great significance for inference.

Therefore, based on the results of the modeling, it is reasonable to believe that most of the new partners will score above 90, although the lowest score is 76.99, which means, that we will reconsider whether we will accept their applications to be our partner because they didn't perform well in the forecast. The average rating in the forecast results is 94.38916, so we feel that the new batch of homestays to be reviewed are excellent overall. We also screened out the variables with high influence on the rating, the most typical ones are price, host response rate, accommodation variable and maximum nights. We hope that more homestays can accommodate more guests, that the owners can respond to questions more quickly, and that they can provide longer stays.

Exploratory Data

Data Overview

We have two datasets in our hands. First, we need to clean the datasets. The data of training and testing are basically the same except the average rating, but the format of some data is inconsistent. When missing values occur in testing, we can't simply remove the line, and we need to process the two data sets in different ways. Here is a detailed explanation of how the different data are processed.

Dispersion

To determine the proper model that can be used in our analysis, we need to have an overview and figure out the degree of dispersion of the data in “Airbnb_Training”. We drew a histogram of average rating in “Airbnb_Training” and found that most of the values are concentrated in the range of 80-100, which does not fit the normal distribution. Then we used the logarithm and reciprocal of this data set and also failed to get the normal distribution.

Time Variables

We noticed that variable “first_review” is shown as 5 digital time. By using “as.Date” function, the origin date is found as 1899-12-30. We decided to transform this format to single year and month because the customers’ evaluation of homestays may have characteristics though months of a year.

Contact Variables

There is a variable called “host_verifications”. We thought this is an important variable that cannot be ignored. This variable is categorical and has a lot of different observations. We speculated that customers care more about the number of contact information of the homestay owner than the type, and we did not have a better way to deal with it. We wrote a function to transform this variable to numeric.

Price Variables

There are two variables called “weekly_price” and “monthly price”. These two variables indicate prices with discount and have a lot of NAs. Moreover, they have multicollinearity. So these two variables must be processed into usable forms. We divided the price by the corresponding number of days, and the difference between the result and the original price is the discount value. In this way, these two variables can be used as normal continuous numeric variables.

Drop Useless Variables

After observing the data set, we determined that the variables belonging to one of these three categories were not helpful for modeling. Only descriptive columns, columns with too much missing data, and columns with too much duplicate data. Therefore, “country”, “experiences_offered”, “host_acceptance_rate”, “house_rules”, “neighborhood_overview”, “summary”, “transit”, “latitude” and “longitude” were removed.

Fill in Missing Values

For Airbnb_Training, if the object is an observation with too many missing values, we delete the corresponding row.

For Airbnb_Testing, we could not delete observations due to prediction. Therefore, we used different methods to fill in different type of missing values. We used the mode to fill the gaps in descriptive variables, and the average to fill the gaps in numerical variables. For abnormally high numerical variables, we turned them into the maximum value in the normal numerical range.

Modeling

kNN

At the beginning, we chose kNN as our starting point because kNN doesn't need assumption and is not sensitive to outliers. We ignored most of the categorical variables and transformed logical variables into dummy variables. After making sure every variable is numeric and scaled, we ran the model.

The main problem encountered in the KNN model is that we cannot determine the value of K, and the way we choose is to reduce the value range of K by dichotomy. Although the accuracy of the KNN model doesn't have a continuous trend of change, the range can be reduced by dichotomy according to the overall trend within a large value range of K. Finally we found that the model did not meet our expectations.

Linear regression (Lasso, Ridge, Stepwise)

After running the kNN model, we want to try the linear regressions. Since the predicted value is not a question of Y=1 or Y=0, we do not consider logistic regression in this question.

Firstly, we ran the basic linear regression, and found that the adjusted R squared was very small, so we considered optimizing the linear regression. After applying the Lasso model, Ridge model and Forward and Backward Stepwise model, we find that the result is more accurate than the KNN model (RMSE is smaller), but we still expect better results.

When reflecting on the problem, we found that there were three myths between these models and data sets. Firstly, In this data set, different variables show strong multicollinearity characteristics. The specific manifestation is that the VIF in the initial model is high. That is to say, there is a high correlation between different variables, which leads to the deviation of the prediction results of the model when these variables are used together for prediction. We tried to classify interaction variables, but it was difficult to avoid this problem completely because there were too many variables. Secondly, when the avg_rating variable in training data is graphically processed, it is obvious that the distribution of predicted values does not show normal distribution, but is more concentrated in the range of 90-100. Linear models perform better for normally distributed data, which may also be one reason why these linear models are not suitable. Overfitting is also one of the problems. Our solution is to delete some outlier data, but the effect is still not ideal.

Tree (Single Tree, Bagging, Random Forest)

After the linear model, we tried the model related to the decision tree. Among single tree, bagging and random forest, the one that finally achieves the satisfactory effect is random forest. When both the data-driven model and the model-driven model do not perform well, "Tree" becomes a scheme worth considering. The advantage of

“Tree” here is that it does not rely entirely on data distribution like the KNN model, nor does it perform better on normally distributed data like the linear model.

The model of a single tree is too simple, and bagging has splitting limitations compared with random forest, so our final adjustment is based on random forest. In the random forest, there are two parameters that need to be determined. One is mtry, one is the number of trees. After several attempts, we determine “mtry=7, tree=481”. It means that the model will take 7 variables from the entire training data set as a reference for each split and there are 481 trees in the entire forest.

Model Evaluation

In these models, we use 30% of the “Airbnb Training” data as the “training data” and the rest of them are used in “testing”. After determining the model, we will put “Airbnb Testing” data into the model for prediction. This time the prediction is a continuous number of variables rather than a variable of Y=0 or Y=1, so that instead of accuracy we use RMSE to measure if the model is good or not. Calculate the difference between the predicted and actual value of the training set, then calculate the RMSE based on this difference and this is our baseline. When the model is used to predict the "testing" portion of the "Airbnb Training" dataset, if RMSE is better than the baseline, our model is useful in forecasting. We expect the model to have the lowest RMSE in all of the models, however we can’t make sure that the best model in predicting the “testing” is also good at predicting the “Airbnb Testing”. We used all the models that performed better than Baseline to predict the "Airbnb Testing" and uploaded the results to the "Kaggle" prediction. Our best result comes from the Random Forest model, and the predicted RMSE is 7.01168.