Hotel Recommendation Algorithms

In documenting my process in determining optimal hotel recommendations, I applied both R and Python throughout the process. Initially, I used Python to decrease the size of the data set since it was causing my system to slow down. In sub-setting the data, I gathered a random sample of about 100,000 rows from the original data set while reading it into a data frame. Once the data set was split, I imported the data frame into a csv for which exploratory data analysis and data preparation will be done in R.

The first step was to search for any missing and null values and remove them from the set. Next, I decided to restructure the date variables since one had included a specific time which I did not believe to be necessary and I wanted to be able to allow the month and year of the variable play in the possibility of recommending a booking. Along with the date variables, I transformed any remaining variables into integers in order to perform the exploratory analysis. Within this analysis, I simply gathered summary of the variables as well as created a histogram to see the distribution of hotel clusters. I also performed a correlation test to determine which variables were strong candidates for feature selection towards the modeling process. Lastly, I merged the two data sets, train.csv and destination.csv, on the destination ID to have one set of information based on the customer’s searches then followed by exporting it to a CSV to be read back into my Python notebook for modeling and further analysis.

Within my Python notebook, I wanted to group the most relevant info in determining where a customer would book a hotel. When traveling, I would assume that people will determine where they would want to go followed by the type of hotel they would want to stay in at the time. Therefore, I grouped these relevant fields and sum the clusters that surrounded each destination ID. With aggregating the sums and counts of these groups into a pivot table for a data frame, it allowed for someone to determine which hotel clusters will be associated. Since we now have our data frame of relevant variables to hotel clusters and the combined data frame of training and destinations, we can now combine them one again on destination search ID to start selecting features and modeling. We will be using the customer identification, their cluster that they belong to, and if they have booked a hotel in the past. The first algorithm I decided to use was the K-Nearest Neighbor Classifier since we are classifying which cluster someone will belong to base on their reviews. At the same time, since we are determining classification, I chose to also use Random Forest Classifier and since it is widely used for learning algorithms. For my third option, I planned on using Multi-Class Logistic Regression incorporating OVR, the one-vs-rest method since we are determining the hotel clusters against the rest of the attributes. Lastly, the choice of using a Support Vector Machine was chosen since it can handle regression and classification tasks but it is typically used for classification objectives.

The accuracy results of each of the four models chosen:

K-Nearest Neighbor Classifier: 61.19%

Random Forest Classifier: 56.91%

Multi-Class Logistic Regression: 65.01

SVM Classifier: 58.75%