**Problem Statement**

Expedia has provided logs of customer behavior. This data includes hotel clusters based on historical price, customer star ratings, geographical locations relative to city center, etc. They are interested in predicting which hotel group a user is going to book. Our goal is to predict the booking outcome for a search.

* **Dataset**

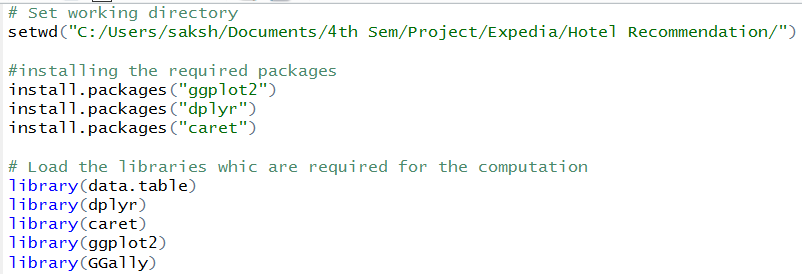
The data that we had was very big, about **4GB**. Below are the column descriptions for the data set.



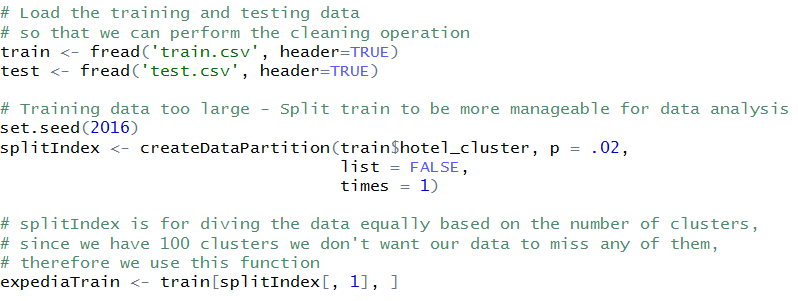


* **R Script**

Initially we are going to set the working directory for the hotel recommendation input file i.e., train/test.csv to clean it. After that we are going to install the libraries like ggplot2, caret, dplyr. After installing we load the libraries.

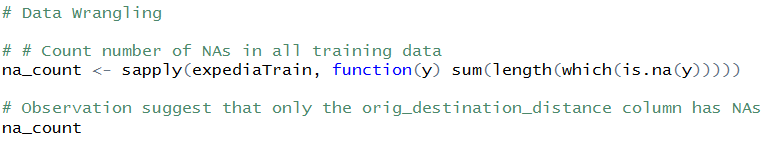


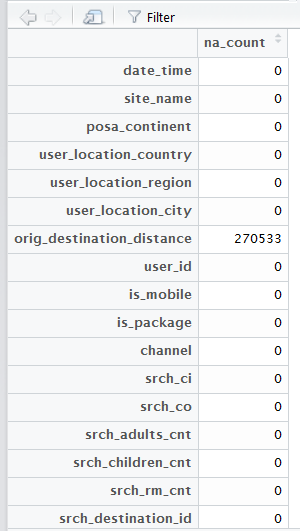
We are loading the training and testing data using fread which is the function in data.table which reads the data very fast compare to read.csv, since our data is approximately 4 GB we need to use this. Since before we proceed to any cleaning we are going to use the 2 percent of the data for data analysis what all columns are there and what are the missing data or outlier, so that we can get the gist of the data. Split Index is for diving the data equally based on the number of clusters, since we have 100 clusters we don't want our data to miss any of them, therefore we use this function. That is how although we are considering only 2 percent of the data but it will have all the cluster equally as they are present in the original data.



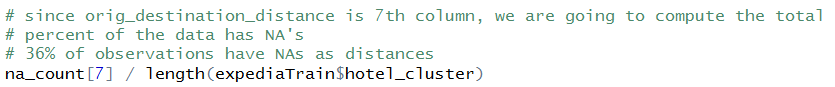
1. **Data Cleaning**

Once we have the 2 percent data, we are going to compute the total NA count in the training dataset.



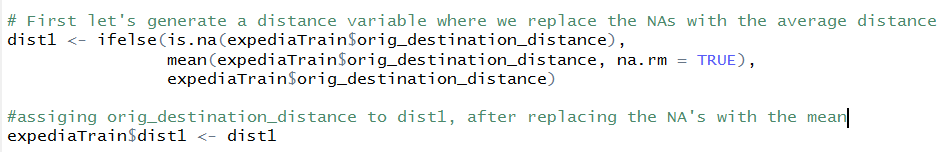


It seems that only the orig\_destination\_distance column has NAs. Now we are going to compute the total percent of dataset has NA’s based on the column orig\_destination\_distance.





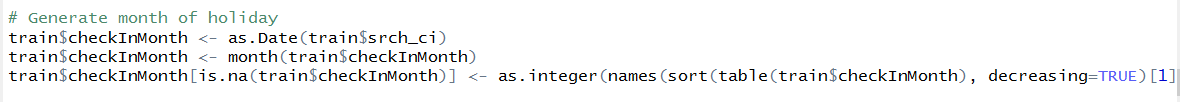
Since we have 36% of the NA in the orig\_destination\_distance, we cannot ignore or remove those rows from the dataset, which is not usual. So therefore in this case, we are going to replace the orig\_destination\_distance with the mean of the column, which is the basic technique for handling missing data and we are going to assign it to new column **dist1**

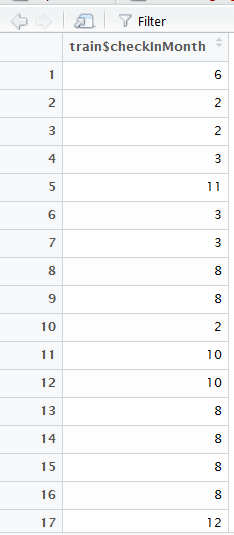


1. **Feature Engineering**

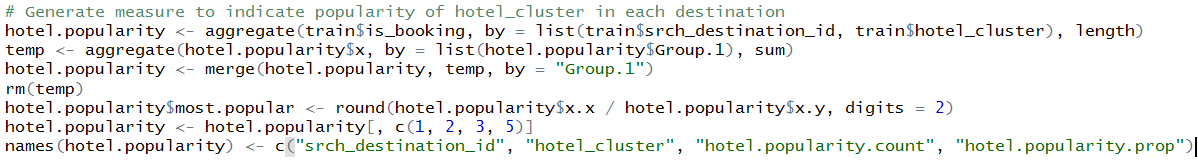
Since we have handle the missing data, and data cleaning is done, we are going to do some feature engineering.

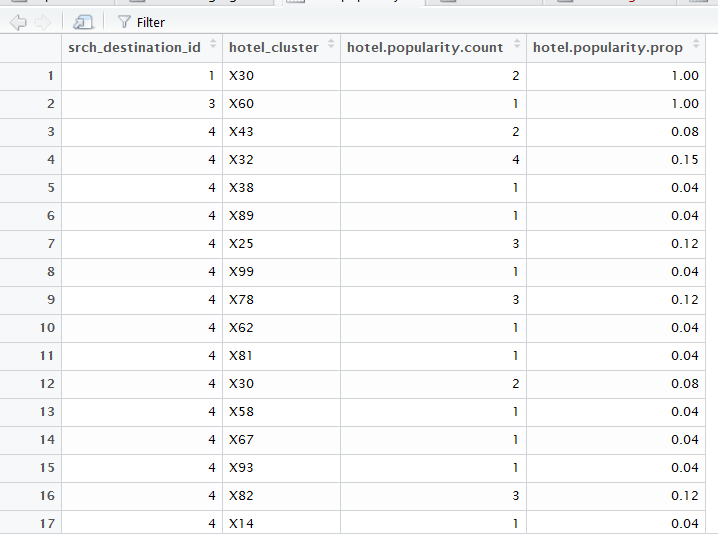
* We are going to find the month of holidays, that is, the month when most of the hotel’s are booked or the month for which most queries are being done.



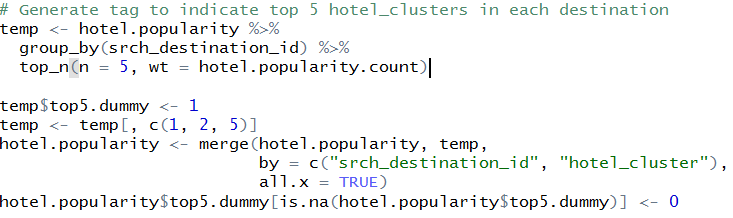


* After getting the month of holiday, we are going to generate the measure to indicate popularity of hotel\_cluster in each destination.





* We are going to generate tag to indicate top 5 hotel\_clusters in each destination

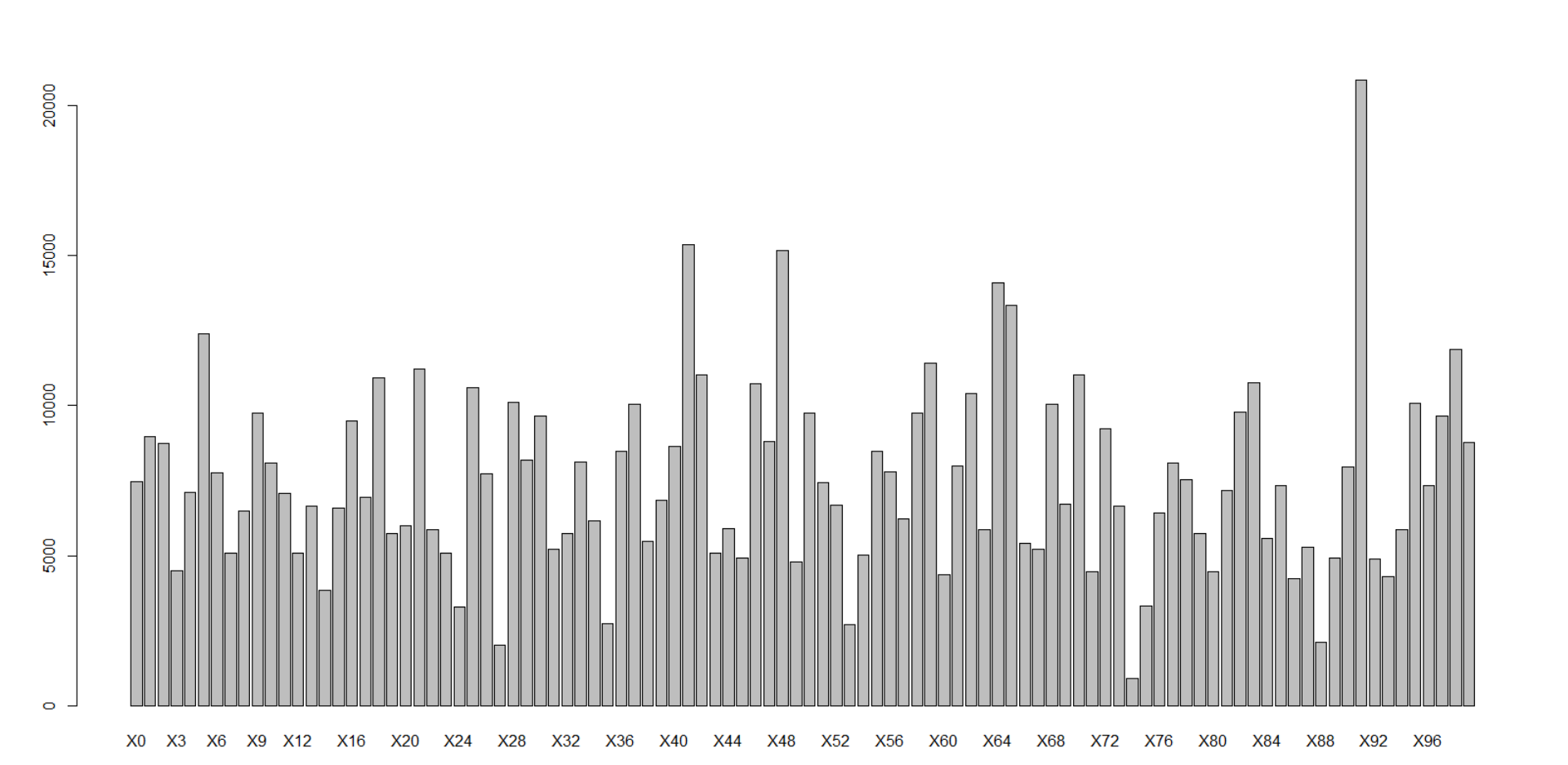


Writing the file to the **expediaData.csv** so that we can use the file for training the model and predicting the values. We are going to implement **Random Forest, Neural Network and Decision Jungle.** The one with the best accuracy is going to be expose to the front end, which is used by the business user to get the prediction.

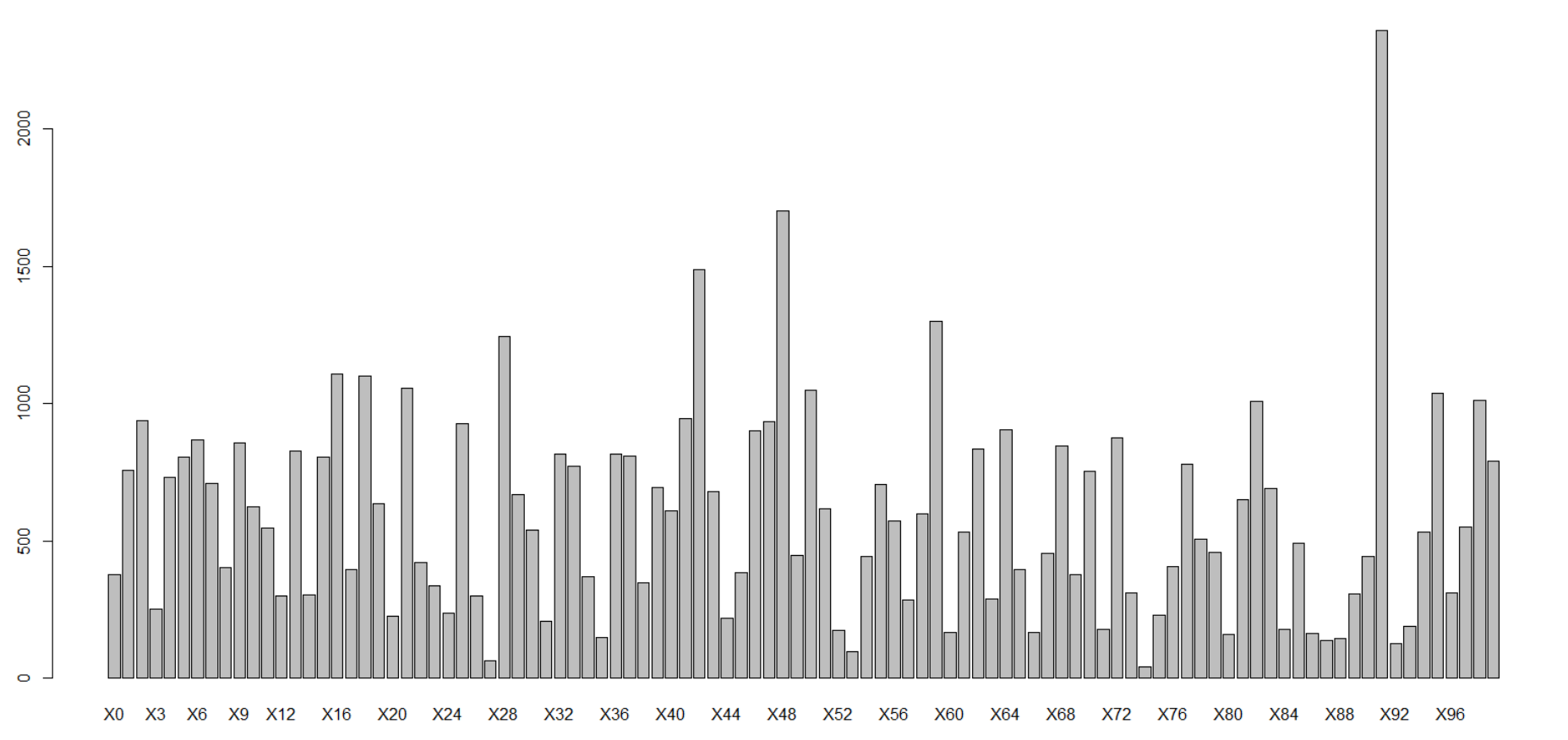


1. **Data Visualization**

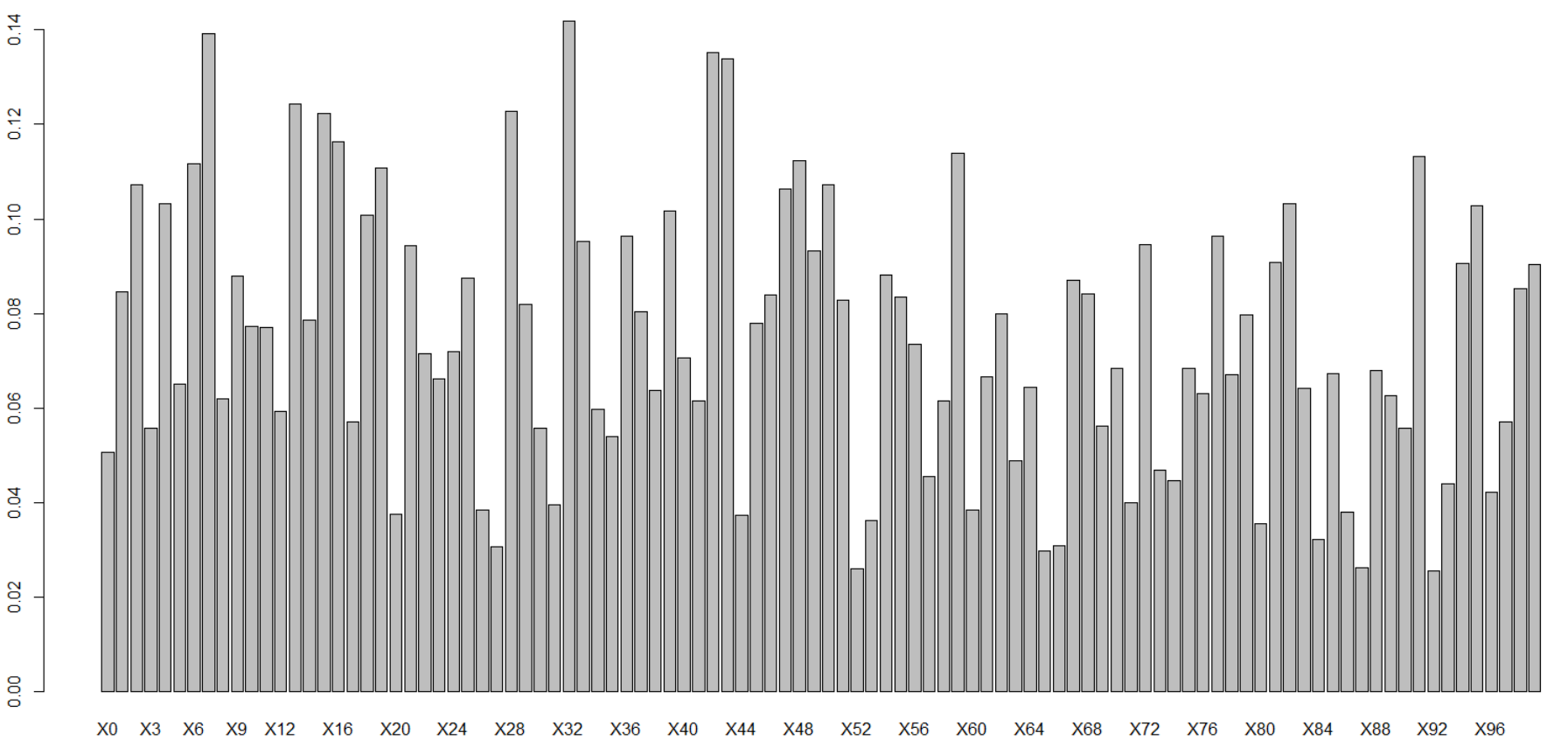
* Plotting the distribution of hotel clusters in training data



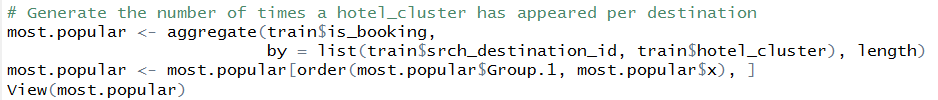
* Count number of times each hotel\_cluster is booked in the training data - to see most popular hotel

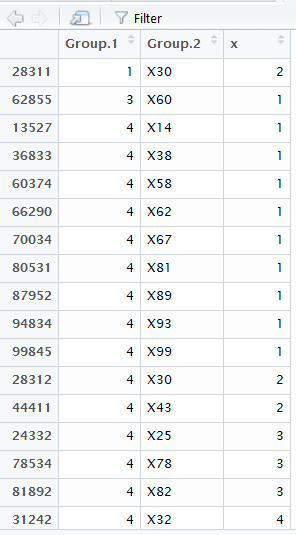


* Looking at most popular hotel\_cluster by looking at % of times that it was booked

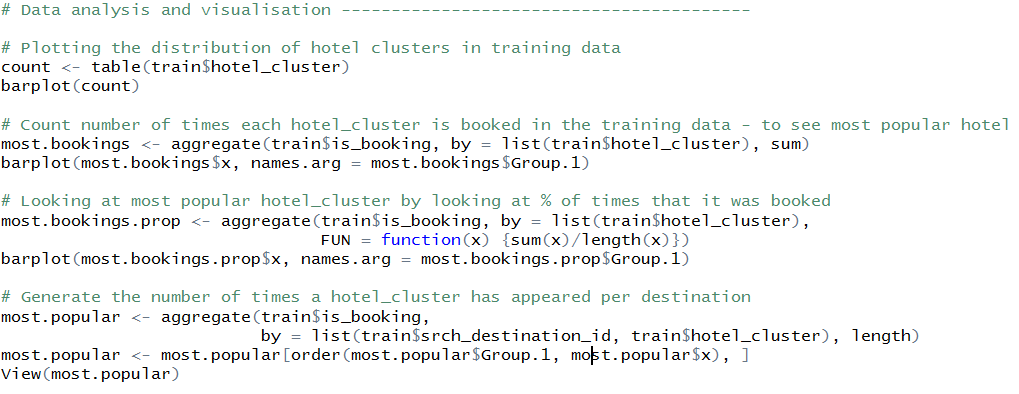


* Generate the number of times a hotel\_cluster has appeared per destination

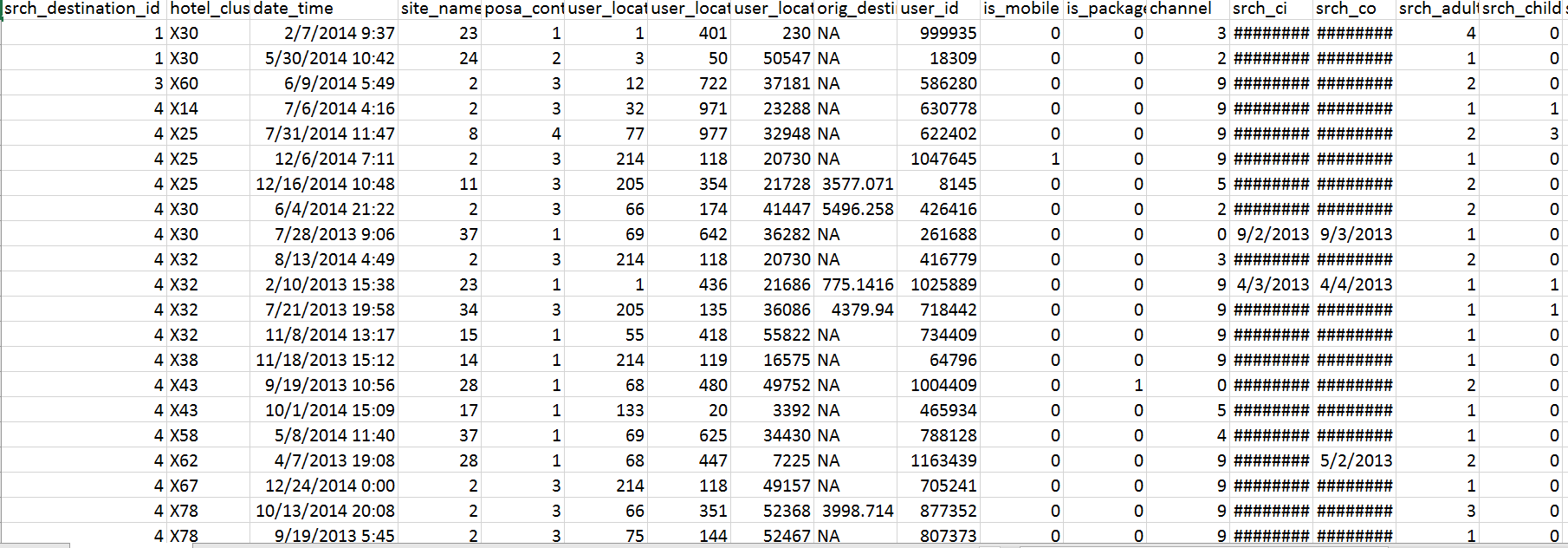


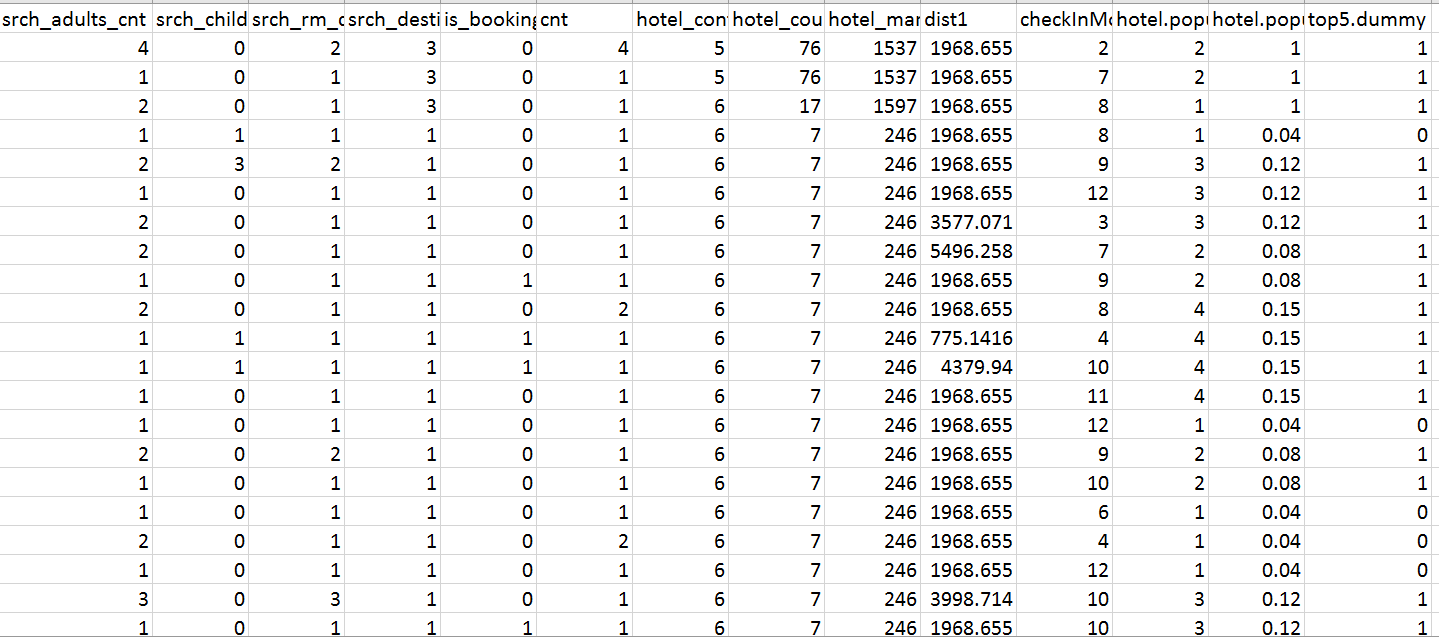


Code for the Data Visualization



* **Clean Train File**



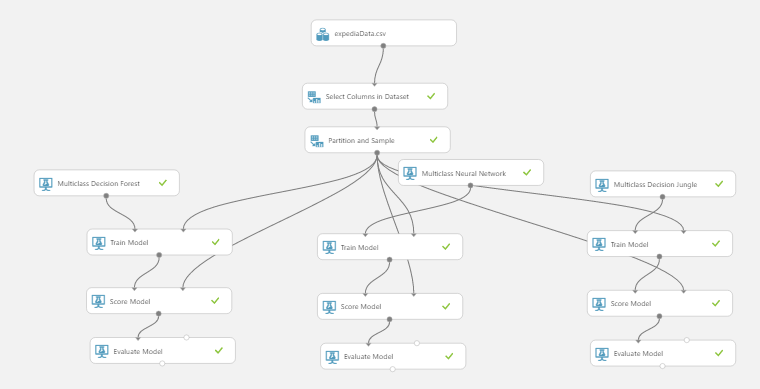


* **Expose as Web Service**

We want to use the prediction in the UI, so we are going to expose the algorithm as a web service to expose for that we are going to use the Microsoft Azure and use the in build algorithms for training the data as well.

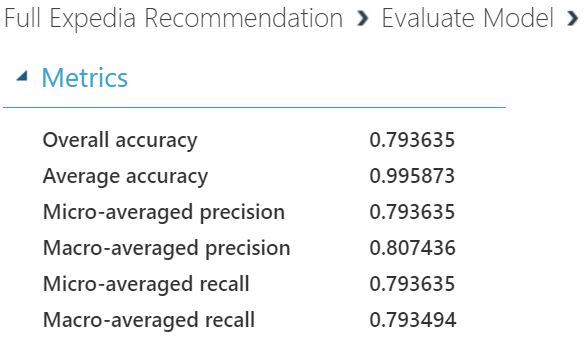
We are using **Mutliclass Classification: Random Forest, Decision Jungle and Neural Network,** later we are going to compare the results and expose the best model to the UI.

First, we are selected the attribute from the dataset, we are not going to use all the attributes, the selected attributes are **distance of destination from the user country, destination ID, hotel market, location attractiveness of hotel, hotel characteristics, user aggregated purchase history and expedia site id.** Once we select the attributes, we can use **Partition and Sample** to split the data into 80% and 20%. Later using Train Model and algorithm we can train the model, after that we can use score model and evaluate model to calculate the performance metric and confusion metrix.

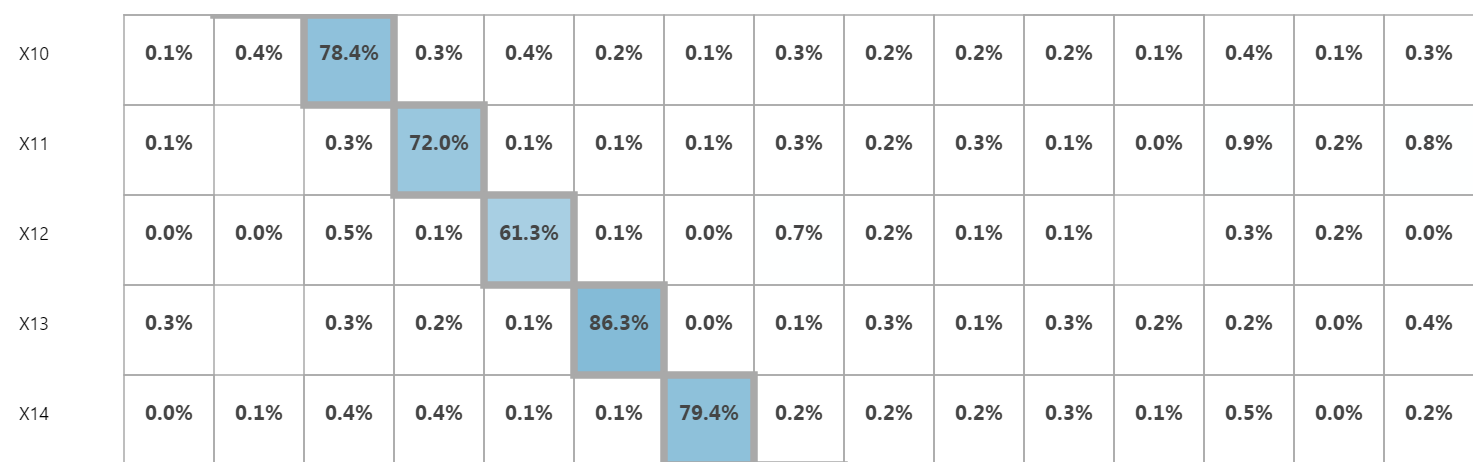


**Random Forest**

**Performance Metric**

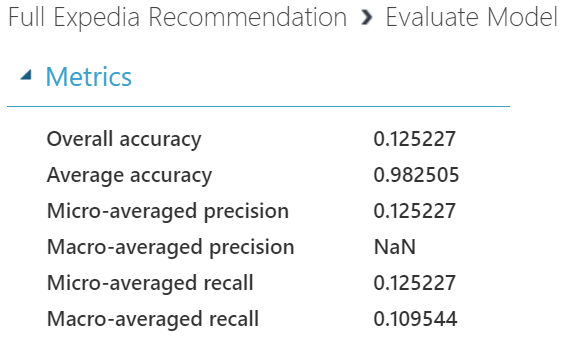


**Confusion Metric**

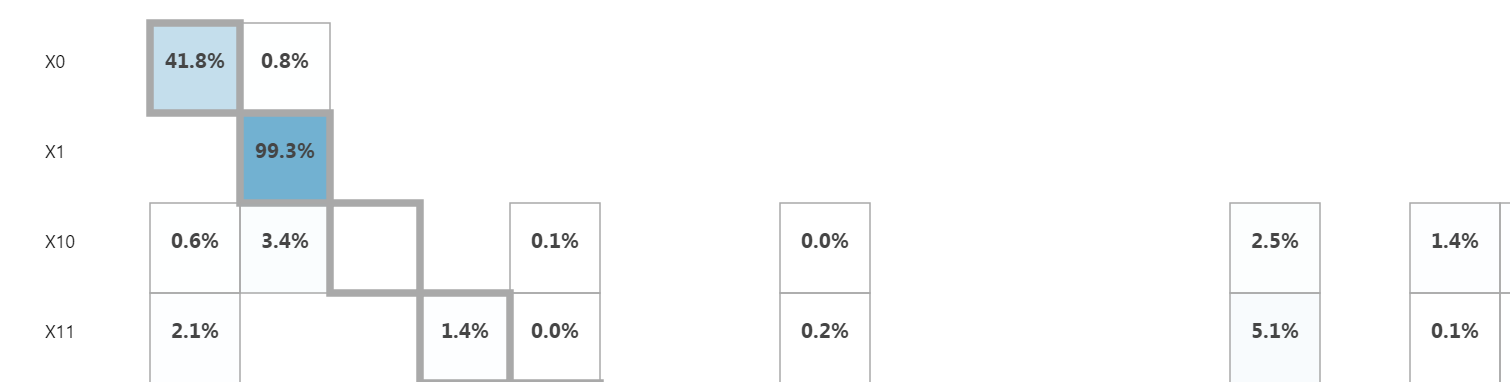


**Decision Jungle**

**Performance Metric**

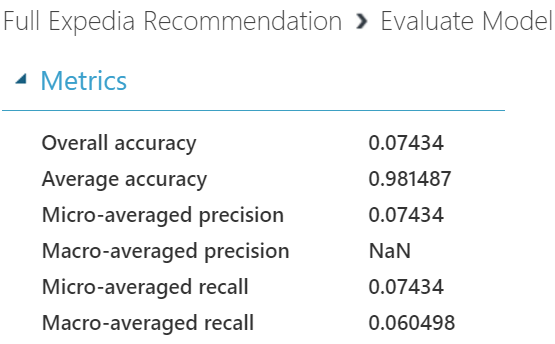


**Confusion Metric**

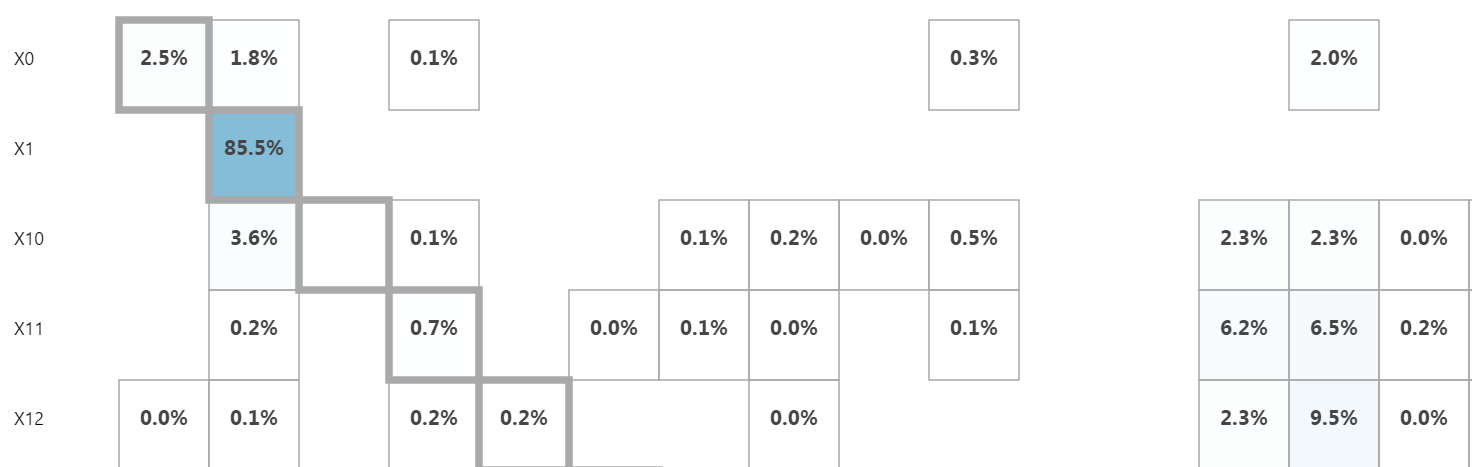


**Neural Network**

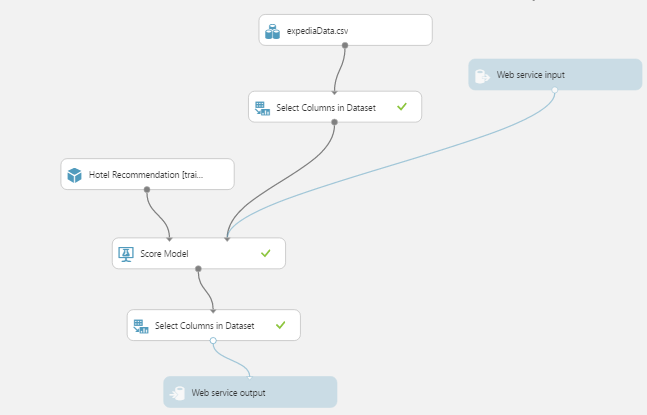
**Performance Metric**



**Confusion Metric**



After observing the performance metrics and confusion metrics of all the three algorithms, we decide to opt for Random Forest which gives us the accuracy of 80%



Web service input and output are used for getting the inputs for the web service and output will give the cluster which is recommended to user.