Project Report
on
Flight Price Prediction

FLIP ROBO

BY:

MAYANK SHUKLA

Background & Introduction

We are about to deploy an ML model for Flight price prediction and analysis. This kind of system becomes handy for many people.

Imagine a situation where you have to book a flight for yourself or for other person, flight price has been changed dyanamically with the time period how soon you booked the ticket its all about the pricing game.

Because of heavy traffic on the bookings on flight seat, the cheap ticket price is a myth for most of the passengers now a days.

If you are planing to a trip or something so you need to book the seat as soon as possible at least a month back we need to book it otherwise there is one more way to do that is, lots of research on the booking site like yatra, makemytrip, paytm and many more for the cheapest price.

Business Problem

Anyone who has booked a flight ticket knows how unexpectedly the prices vary. The cheapest available ticket on a given flight gets more and less expensive over time. This usually happens as an attempt to maximize revenue based on - 1. Time of purchase patterns (making sure last-minute purchases are expensive) 2. Keeping the flight as full as they want it (raising prices on a flight which is filling up in order to reduce sales and hold back inventory for those expensive last-minute expensive purchases) So, you have to work on a project where you collect data of flight fares with other features and work to make a model to predict fares of flights.

Approach to Data Cleaning

* First we check the information of the given dataset because it tells that how many rows and columns are present in our dataset and data type of the columns whether they are object, integer or float.

- Drop duplicates rows if present in dataset.+Then we check for the null values present in our dataset.
- If null values are present then fill it via mean, median or mode. Or also you can remove that rows but kindly check it properly.
- After that we check the summary statistics of our dataset. This part tells about the statistics of our dataset i.e. mean, median, max value ,min values and also it tell whether outliers are present in our dataset or not.

Approach to Data Cleaning (Cntd…)

* We also check the correlation of our dataset to check the correlation of the columns with each other. If columns are highly correlated with each other let’s say 90% or above then remove those columns to avoid multi coli-nearity problem.

* We extract data from date column and make new columns like day, month and year to see the outcomes with our target column that is label.

* We delete the pcircle column because it has only one unique value that tells that collected data is only for one circle.

* We cannot remove outliers because more than 20% of our data are removed.

Visualization.

* We plot correlation matrix via heatmap to see the correlation of the columns with other columns.

* We also visualize the correlation of columns with target column via bar graph to see which column is highly correlated with target column.

* We see the number of defaulter and non defaulter customers with the help of count plot.

* We plot histogram to displays the shape and spread of continuous sample data.

* We also see the customers labels i.e defaluter/Non-defaulter according to date and month with count plot.

* We also see the distribution of the data with the help of distribution plot whether it is left skewed or right skewed.

Modelling part

* We know that this is classification problem so we use accuracy score, classification report and confusion matrix as our evaluation matrix. We also see the AUC score and also plot the AUC_ROC curve for our final model.

* As we know this dataset is imbalance so we don't too much focus on accuracy score . We see the precision and recall value along with f1_score.

* First we see the result without doing any sampling technique and for that I Linear Regression Model

* And then we use KNeighborsRegressor where the input consists of the k closest training.

* At the third time we use DecisionTreeRegressor model, which is a model of decisions and all of their possible results, including outcomes, input costs, and utility.

* Then we use RandomForestRegressor Model, which performing both regression and classification tasks with the use of multiple decision trees and a technique called Bootstrap and Aggregation, commonly known as bagging.

* Finally we use AdaBoostRegressor and GradientBoostingRegressor model, which is used to generate an ensemble model by combining the weak learners or weak predictive models.

Conclusion

So here '**KNeighborsRegressor**' is the best model out of all model tested above which gives score - 0.588625 which is the greatest.