**Amplify Analytix Task**

**Problem Statement:**

The data is from one of the world’s largest online travel agencies (OTA) which powers search results for millions of travel shoppers every day.

In this competitive market matching users to hotel inventory is very important since users easily jump from website to website. As such, having the best ranking of hotels (“sort”) for specific users with the best integration of price competitiveness gives an OTA the best chance of winning the sale.

The dataset includes shopping and purchase data as well as information on price competitiveness. The data are organized around a set of “search result impressions”, or the ordered list of hotels that the user sees after they search for a hotel on the agency's website. The user response is provided as a click on a hotel or/and a purchase of a hotel room.

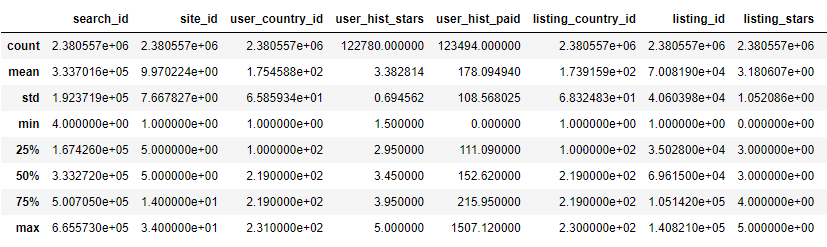
To predict if the user chooses a hotel based on the users’ browsing history from search to bookings and payments.

**Exploratory Data Analysis and Data Pre-processing:**

**Types of features in the dataset**

|  |  |
| --- | --- |
| search\_id | float64 |
| timestamp | object |
| site\_id | int64 |
| user\_country\_id | int64 |
| user\_hist\_stars | float64 |
| user\_hist\_paid | float64 |
| listing\_country\_id | int64 |
| listing\_id | int64 |
| listing stars | int64 |
| listing\_review\_score | float64 |
| is\_brand | int64 |
| location\_score1 | float64 |
| location\_score2 | float64 |
| log\_historical\_price | float64 |
| listing\_position | int64 |
| price\_usd | float64 |
| has\_promotion | int64 |
| destination\_id | int64 |
| length\_of\_stay | int64 |
| booking\_window | int64 |
| num\_adults | int64 |
| num\_kids | int64 |
| num\_rooms | int64 |
| stay\_on\_saturday | int64 |
| log\_click\_proportion | float64 |
| distance\_to\_dest | float64 |
| random\_sort | int64 |
| competitor1\_rate | float64 |
| competitor1\_has\_availability | float64 |
| competitor1\_price\_percent\_diff | float64 |
| competitor2\_rate | float64 |
| competitor2\_has\_availability | float64 |
| competitor2\_price\_percent\_diff | float64 |
| competitor3\_rate | float64 |
| competitor3\_has\_availability | float64 |
| competitor3\_price\_percent\_diff | float64 |
| competitor4\_rate | float64 |
| competitor4\_has\_availability | float64 |
| competitor4\_price\_percent\_diff | float64 |
| competitor5\_rate | float64 |
| competitor5\_has\_availability | float64 |
| competitor5\_price\_percent\_diff | float64 |
| competitor6\_rate | float64 |
| competitor6\_has\_availability | float64 |
| competitor6\_price\_percent\_diff | float64 |
| competitor7\_rate | float64 |
| competitor7\_has\_availability | float64 |
| competitor7\_price\_percent\_diff | float64 |
| competitor8\_rate | float64 |
| competitor8\_has\_availability | float64 |
| competitor8\_price\_percent\_diff | float64 |
| clicked | int64 |
| booking\_value | float64 |
| booked | int64 |

**Summary statistics**



**Handling missing values:**

List of features with missing values

|  |  |
| --- | --- |
| ‘user\_hist\_stars' | 'competitor3\_price\_percent\_diff' |
| 'user\_hist\_paid' | 'competitor4\_rate' |
| 'listing\_review\_score' | 'competitor4\_has\_availability' |
| 'location\_score2' | 'competitor4\_price\_percent\_diff' |
| 'log\_click\_proportion' | 'competitor5\_rate' |
| 'distance\_to\_dest' | 'competitor5\_has\_availability' |
| 'competitor1\_rate' | 'competitor5\_price\_percent\_diff' |
| 'competitor1\_has\_availability' | 'competitor6\_rate' |
| 'competitor1\_price\_percent\_diff' | 'competitor6\_has\_availability' |
| 'competitor2\_rate' | 'competitor6\_price\_percent\_diff' |
| 'competitor2\_has\_availability' | 'competitor7\_rate' |
| 'competitor2\_price\_percent\_diff' | 'competitor7\_has\_availability' |
| 'competitor3\_rate' | 'competitor7\_price\_percent\_diff' |
| 'competitor3\_has\_availability' | 'competitor8\_rate' |
| 'booking\_value' | 'competitor8\_has\_availability' |
| 'competitor8\_price\_percent\_diff' |  |

All the missing values have been handled by replacing with:

* median if the feature is skewed
* mode if it contains categorical data
* mean if it follows normal distribution

**Data Sampling and Normalization:**

* Used Stratified split to select 1% of the data from the original dataset and to ensure the class balance is kept intact.
* Normalized the features which had higher range to bring it to a uniform scale.

**Model details**

I have choosen logistic regression model as the class variable is categorical in nature. Initial model was ran by considering all the parameters. This resulted in a low f1 score. Hence feature selection and hyper parameter tuning was applied to increase the score.

**Feature selection**

1. Used corr() for detecting multicollinearity among features.

* In case of multicollinearity, we might lose reliability in determining the effects of individual features in the model and hence, we remove these features.
* To resolve this, we use corr() method which returns a correlation score for each feature.
* This score determines the strength of correlation between the independent variables. All the features with a score greater than 0.50 were removed from the dataset suggesting high multicollinearity.

The following features: search\_id, listing\_id and time\_stamp are irrelevant and hence have been removed.

**Model Building and Hyperparameter tuning**

1. Choosing the right predictor variable:

Out of the 3 possible outcome variables: clicked, bookings\_value and booked. I chose "clicked" as the outcome variable for my model considering the following reasons:

1. Since, we are trying to predict whether the user is going to pick the hotel given multiple choices of hotels and competitors’ websites, the ideal choice would be to predict the maximum probability of a click (“clicked”) which indicates a visitor’s interest and potentially a decision to book.
2. “Bookings\_value” does not apply here since it does not address the problem statement in hand, our goal is not price prediction rather if the user is going to pick a hotel given multiple choices of hotels and competitor’s websites.
3. The outcome variable "Booked" would come into picture when we are trying to predict if the user is going to book a particular hotel or not given a single website/source. In our case, we are also trying to compare between different websites, hence the outcome variable "Booked" is not applicable here.

2. Machine Learning algorithm used:

Applied "Cost Sensitive/Weighted Logistic Regression" taking into consideration the following reasons:

* Predicted variable is categorical
* Class has imbalanced data:The dataset has imbalanced class with approximately:

94% - negative class and 6% - positive class because of which the accuracy metrics get inflated. This has been handled by considering the metric "F1 score" while tuning hyper parameters and measuring the success of the model.

3.Hyperparameter Tuning:

* Applied GridSearchCV to fetch optimum values for weights and alpha.
* Used Kfold cross validation to ensure we are not overfitting the model.

4.Selection of significant features:

* Used RFE algorithm to find out significant features for the model and removed insignificant features.

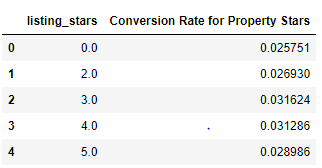
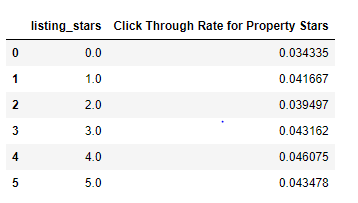
4. Evaluation metrics:

I chose “average f1 score” to evaluate the accuracy of the model and also to tune the hyper parameters of the model.

Since we are predicting class labels and both false negatives and false positives are equally costly, we use the F1 measure as the evaluation metric for our model.

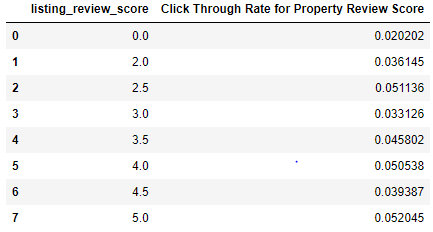
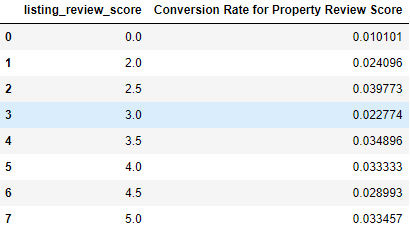
5. Click through rate and conversion rate for property stars:

From the below prediction, we can see that CTR for property stars is highest for 4-star hotel and conversion rate is highest for 3-star hotel



6. Click through rate and conversion rate for property review score:

From the below prediction, we can see that CTR for property review score is highest for score 5 and conversion rate is highest for 2.5 score.

**Conclusions:**

I achieved a model accuracy of 89%. Click through rate and conversion rate have been calculated for different levels of property stars and property review score for the test data.