Airline Passenger Referral Prediction

Content:

1. Introduction
2. Abstract
3. Problem Statement
4. Steps Involved
5. Algorithms
6. Confusion Matrix
7. Interpretability of Model
8. Conclusion

Introduction:

- Predicting airline passenger referral is one of the most important business challenges.
- This is performed by analyzing data using EDA to extract and analyze essential features such as cabin service, food beverage, traveller type, and entertainment, among other things.
- We must create a forecasting model to predict the outcomes based on our problem use case. This would allow them to reward the passenger who made the referral with additional benefits or rewards.
- After the primary objective has been met. They may now anticipate any passenger reviews and carry out the task while studying the results and expanding their business needs.
- Now the Aviation Industry results in increasing their growth.

Abstract:

- Using exploratory data analysis, data wrangling, visualization, and other approaches, we can extract a lot of information from the analysis.From this analysis we can able to get business understanding of the problem.
- we've determined that our primary goal is to predict if a tourist will enthusiastically suggest his or her experience to family and friends.
- We were able to extract the most important aspects for our situation. The majority of the variables are associated with airline ratings, which are critical to our forecasting algorithm.
- For our classification task, we need to create a prediction model. The best model must be built with the help of a number of machine learning algorithms and hyperparameter tuning in order to produce the most effective and realistic results.

Problem Statement:

- From 2006 to 2019, this data includes airline reviews for popular airlines all over the world.
- The first goal is to perform some EDA in order to gain a better understanding of the situation from a business perspective.
- Following our research, we've established that our major goal is to develop a model that can predict whether or not a traveller will enthusiastically suggest his or her trip to family, friends, and the public in general.

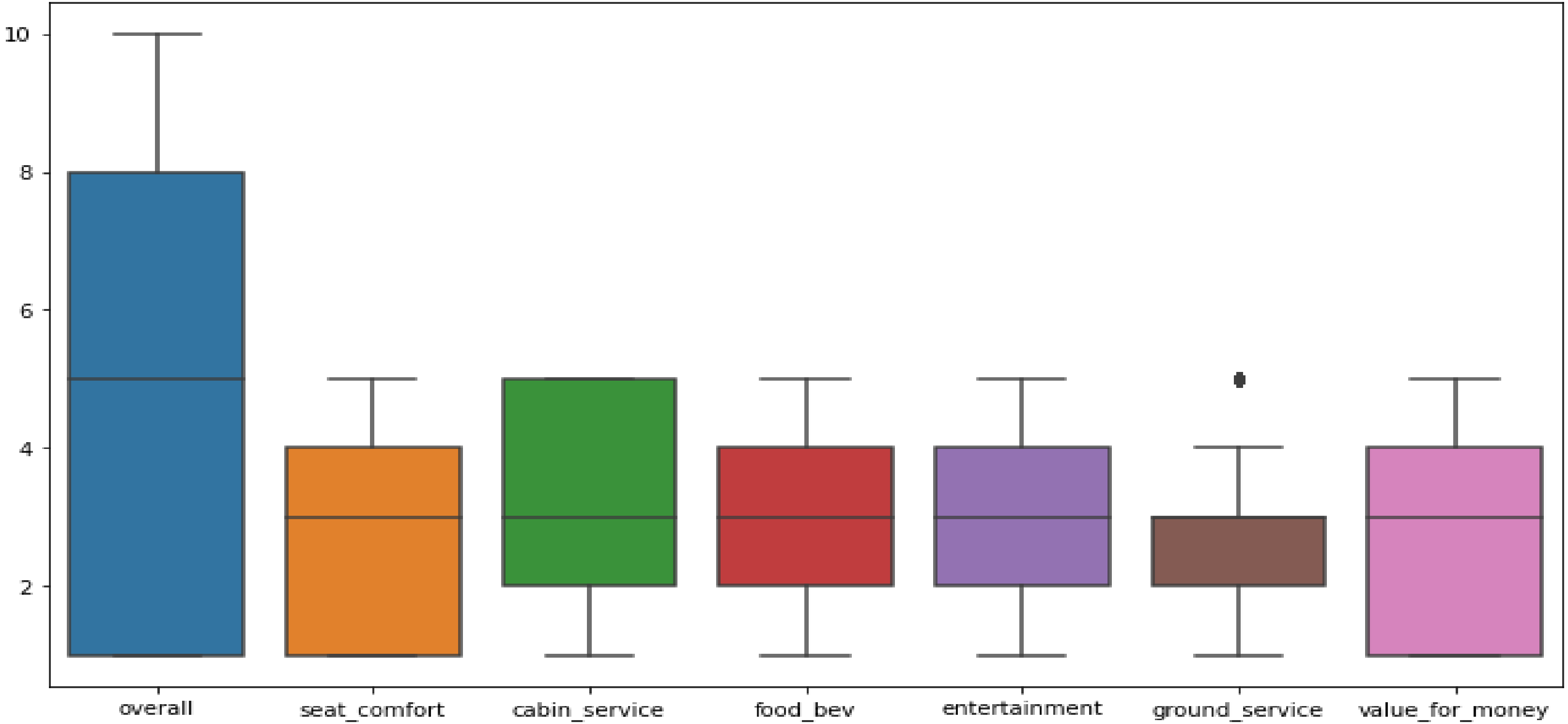
Steps Involved:

- Null values Treatment and Outliers
- Exploratory Data Analysis
- Correlation Analysis
- Label encoding
- Train test Split
- Fitting different models
    - Logistic Regression Model
    - Decision Tree Model
    - Random Forest Model
    - Gradient Boosting Model
- Tuning the hyperparameters for better accuracy

Null values treatment and outliers:

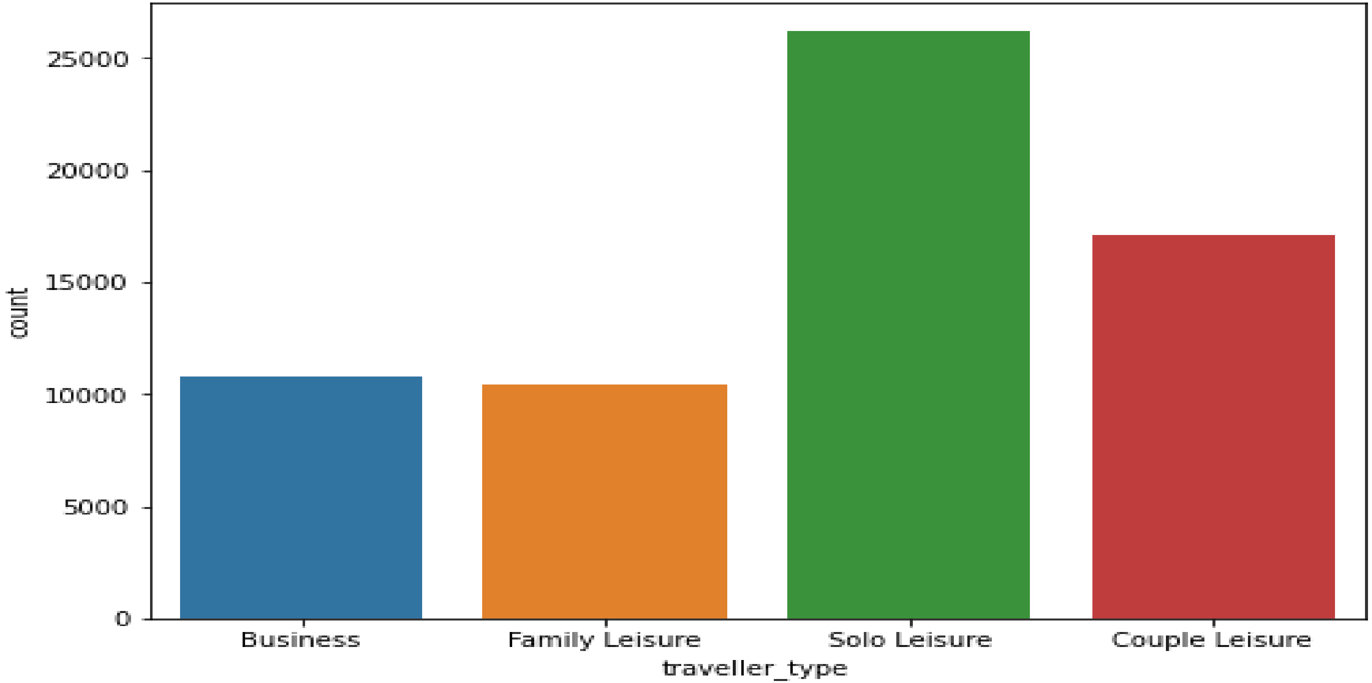
- First, we loaded our dataset, and then we checked the size of it, which has 131895 rows and 17 columns.
- Then we dropped some unneeded columns and checked for null values, using imputation techniques to fill null values in numerical columns and forward fill and mode imputation methods for categorical columns.
- After that, we checked for outliers in each column and found that none existed.
- We can see that there are no outliers in the plot below, hence there is no need to handle outliers.

Outlier plot…



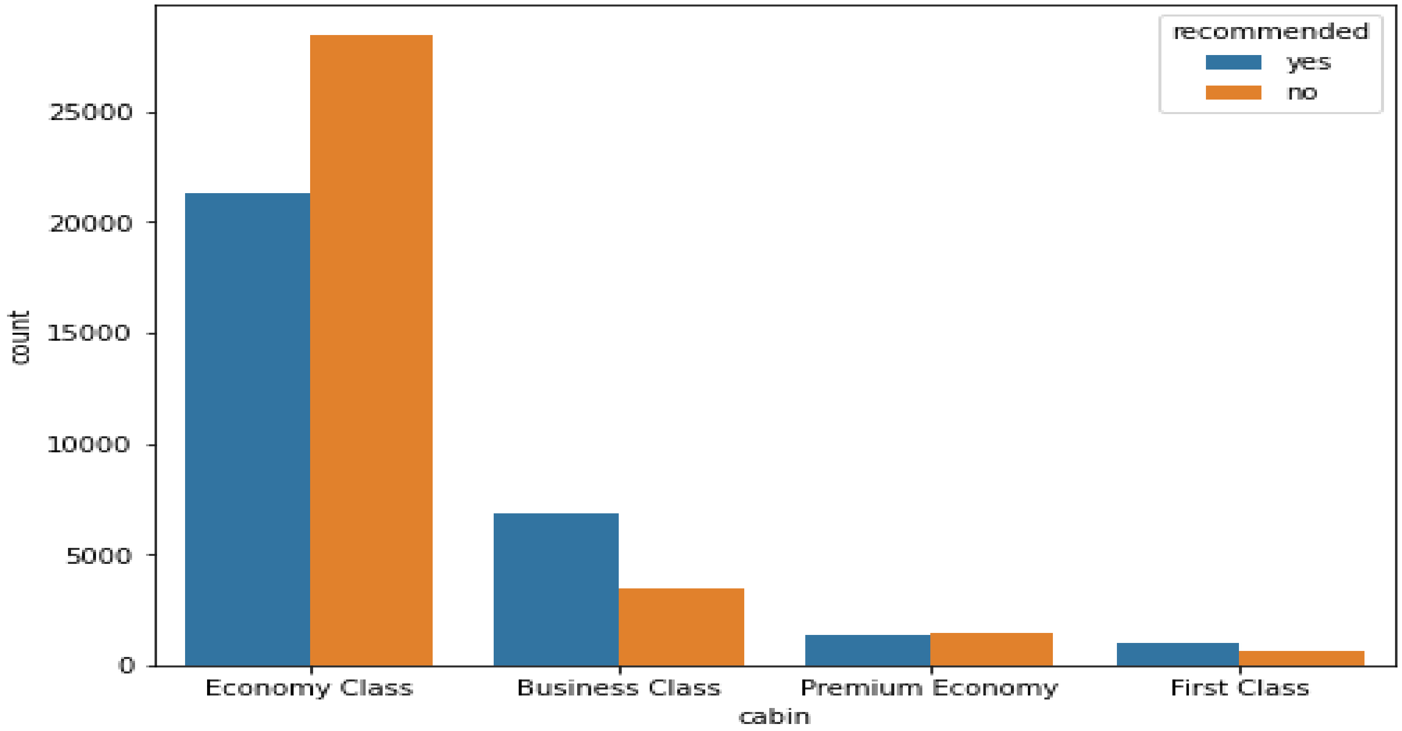
Exploratory Data Analysis:

Analyzing which traveller type has more ratings



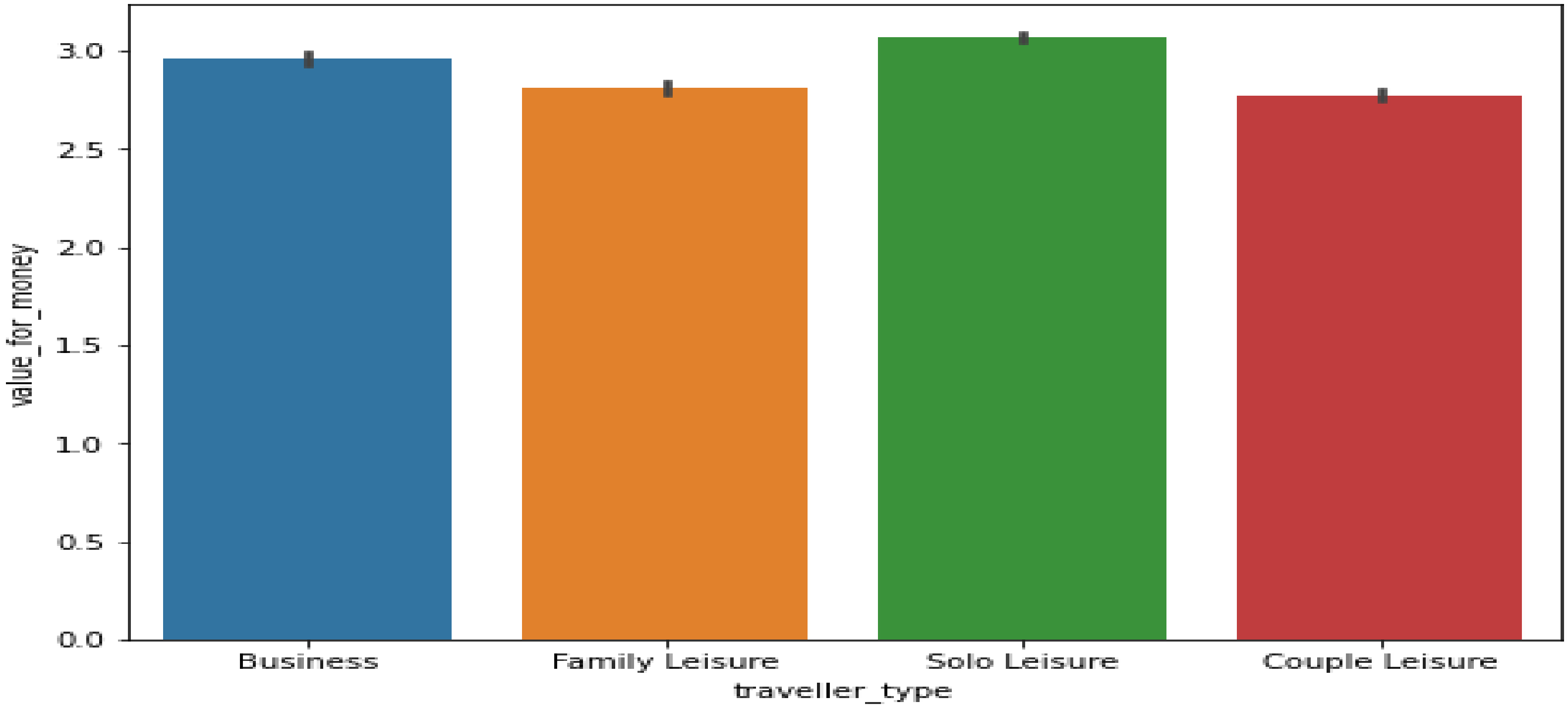
*From above plot we can say that Travelling type of **Solo Leisure** has more ratings*

Analyzing which which type of cabin has more recommendation



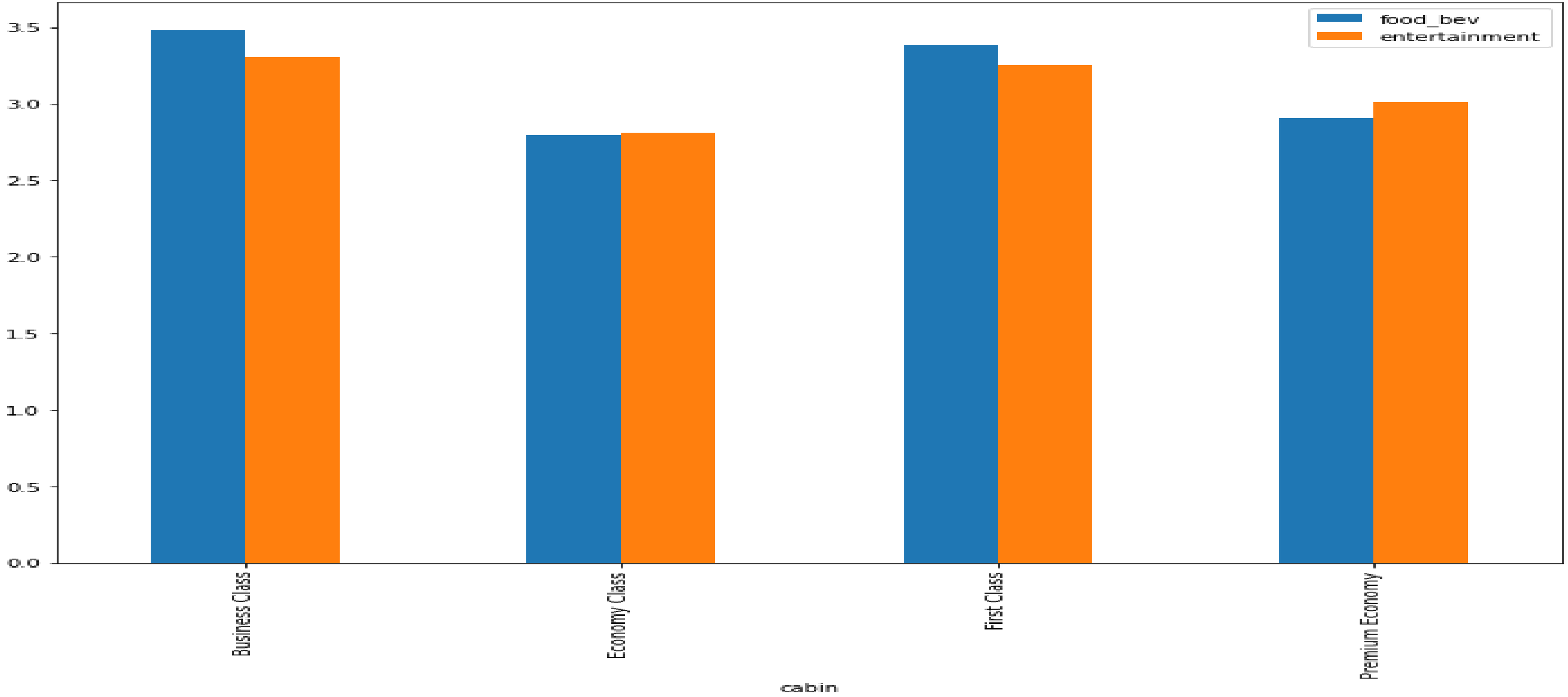
From above countplot we can say that ***Economy class*** *has more recommendation.*

Analyzing solo leisure is worth for money or not?



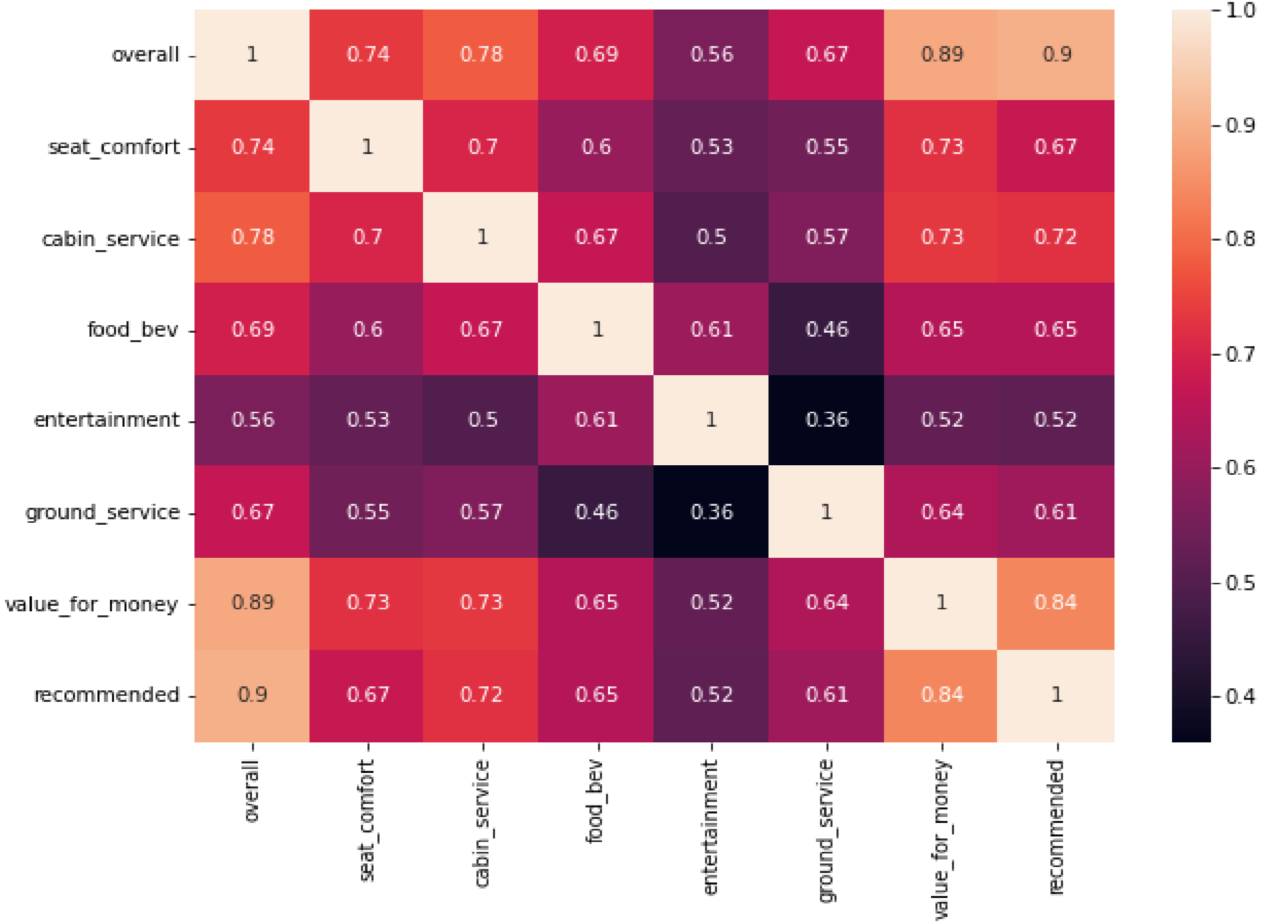
*From above plot we can say that **Yes, Travelling** Type of **Solo Leisure worth of Money** compare to other type of travelling.*

Analyzing average ratings of food bev and entertainment in economy class



*From above plot we can say that In Economy Class the **average ratings** of Food bev and entertainment given by passenger is **lowest** compared to other cabin classes.*

Correlation Analysis:



	overall	seat_comfort	cabin_service	food_bev	entertainment	ground_service	value_for_money	recommended
overall	1	0.74	0.78	0.69	0.56	0.67	0.89	0.9
seat_comfort	0.74	1	0.7	0.6	0.53	0.55	0.73	0.67
cabin_service	0.78	0.7	1	0.67	0.5	0.57	0.73	0.72
food_bev	0.69	0.6	0.67	1	0.61	0.46	0.65	0.65
entertainment	0.56	0.53	0.5	0.61	1	0.36	0.52	0.52
ground_service	0.67	0.55	0.57	0.46	0.36	1	0.64	0.61
value_for_money	0.89	0.73	0.73	0.65	0.52	0.64	1	0.84
recommended	0.9	0.67	0.72	0.65	0.52	0.61	0.84	1

Here we checked **VIF** and **Overall column** and **Value for money column** this two columns got removed due to **multi-collinearity** between them.

Label Encoding:

- As we all know that Label Encoding refers to converting the labels into a numeric form so as to convert them into the machine-readable form.
- Machine learning algorithms can then decide in a better way or understand how those labels can be operated.
- It is an important pre-processing step before building our model.
- We had two categorical variables: traveller type and cabin, which we turned into dummy variables such as Couple Leisure, Family Leisure, and Solo Leisure, and cabin into dummy variables such as Economy Class, First Class, and Premium Economy.

Feature Selection:

- In Feature selection we remove non-informative or redundant predictors from the model.
- At the beginning we had 131895 rows and 17 columns and after feature selection and label encoding we got 64440 rows and 12 columns.

Algorithms:

- We've done null value treatment, removed unnecessary columns, removed multicollinearity features, fitted our independent features in x and dependent features in y variables.
- We split the data into two parts: 80 % for training and 20 % for testing before fitting in to model.
- We have to predict airline passenger referrals. From our problem statement and business use case, Recall is given top priority, accuracy is given second priority, and ROC AUC are given third priority in the classification metrics.

1)Logistic regression:

- The logistic regression algorithm was the first algorithm we implemented.
- We now provided the training data into our Logistic Model for Training.
- Once training is completed we provided the testing data for evaluation of our model.

Metrics	Training Accuracy	Testing Accuracy
Accuracy Score	89%	89%
Recall Score	88%	88%
ROC-AUC Score	89%	89%
Precision Score	89%	89%

2)Decision Tree Classifier:

- We now provided the training data into our Decision Tree Model for Training.
- Once training is completed we provided the testing data for evaluation of our model.
- We are now using a Decision tree classifier to get better recall and accuracy than logistic regression model.

Hyperparameter Tuning For Decision Tree Classifier using grid search cv:

- Our model is overfitted in this case before applying Hyperparameter tuning to Decision Tree. Hyperparameter tuning is used to prune a Decision tree in order to preserve the Generalized Model.
- Once Hyperparameter tuning is performed, Now our Model becomes a Generalized Model.

Metrics	Training Accuracy	Testing Accuracy
Accuracy Score	89%	89%
Recall Score	86%	86%
ROC-AUC Score	89%	89%
Precision Score	92%	91%

Still, we were not able to achieve better recall and accuracy than logistic regression.

3)Random Forest Classifier:

- We now provided the training data into our Random Forest Model for Training.
- Once training is completed we provided the testing data for evaluation of our model.
- We are now using a Random Forest classifier to get better recall and accuracy than logistic regression and Decision Tree model.

Hyperparameter Tuning For Random Forest using grid search cv:

- We couldn't surpass logistic regression recall and accuracy with the previous algorithm. To gain control over the process, we are now using random forest algorithm with hyperparameter tuning using grid search CV.

- Once Hyperparameter tuning is performed.we get a best parameters. They are
    - maximum depth:7
    - minimum samples leaf:2
    - minimum samples split:5
    - numbers of estimators: 80
- Now Fitting these best parameters into our Random Forest Model.

Metrics	Training Accuracy	Testing Accuracy
Accuracy Score	90%	90%
Recall Score	86%	86%
ROC-AUC Score	89%	89%
Precision Score	92%	92%

- We now have a better accuracy result in this model than in logistic regression, but we were unable to achieve a better recall result in this model, therefore we are rejecting it as well.

4)Gradient Boosting Classifier:

- We now provided the training data into our Gradient Boosting Model for Training.
- Once training is completed we provided the testing data for evaluation of our model.
- We are now using a Gradient Boosting Model to get better recall and accuracy than previous model we implemented.

Hyperparameter Tuning For Gradient Boosting using grid search cv:

- We had better accuracy than logistic regression in the previous algorithm, but we weren't able to have better recall than logistic regression, therefore we're now implementing gradient boosting algorithm with hyperparameter tuning using grid search cv to gain control over the process.

- Once Hyperparameter tuning is performed.we get a best parameters. They are
    - ❖ maximum depth:5
    - ❖ minimum samples leaf:3
    - ❖ minimum samples split:5
    - ❖ numbers of estimators: 80
- Now Fitting these best parameters into our Gradient Boosting Model.

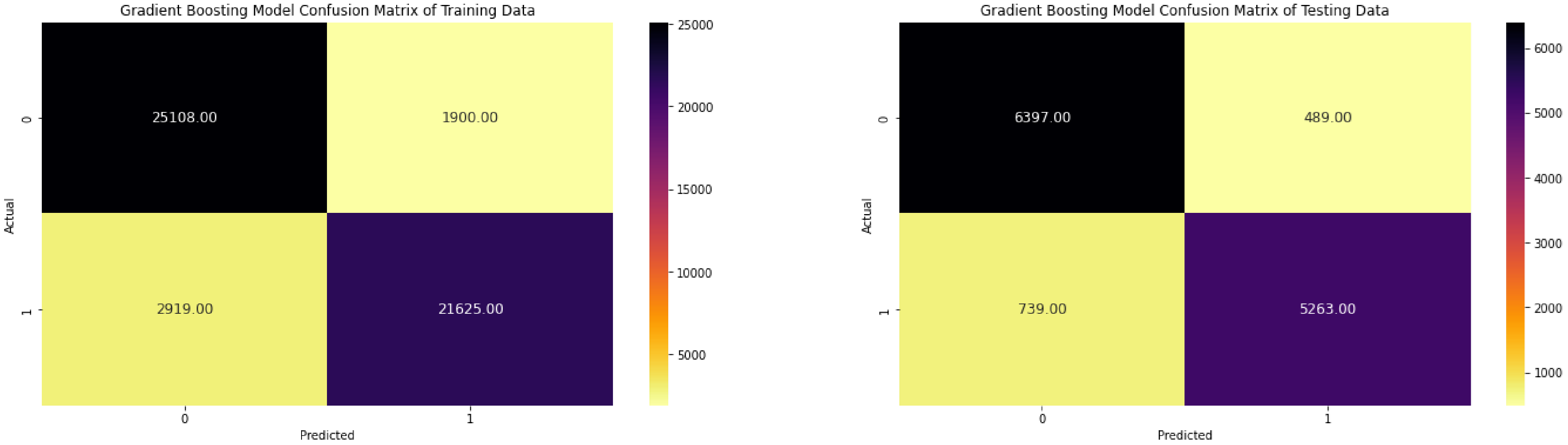
Metrics	**Training Accuracy**	**Testing Accuracy**
Accuracy Score	90%	90%
Recall Score	88%	87%
ROC-AUC Score	90%	90%
Precision Score	91%	91%

- From this algorithm we got better results in terms of **Accuracy Score** than in logistic regression and other Models.
- But in terms of **Recall Score** it is approximately same for both **Logistic Regression** and **Gradient boosting** for both training and testing data.
- Now considering the **ROC_AUC Score** for Gradient Boosting, It is better than all Models which we have implemented previously before.
- From this Metrics,Now we can Finalize that ***Gradient boosting Classifier Model*** will be ***best suitable for predicting Airline referral*** *and* ***best model for Aviation Industry for the given problem***.

Confusion Matrix:

- It is a matrix form or table form that shows how well the classification model predicts which samples belong to which classes. All classification metrics for evaluating a model are rooted in the Confusion matrix. Because this confusion matrix is the origin of many classification metrics.
- As previously stated, we used our confusion matrix to determine accuracy, precision, recall score, and roc auc score.
- To calculate **Accuracy** from confusion matrix we use this formula: **(TP+TN)/(TP+TN+FP+FN)**
- To calculate **Precision** from confusion matrix we use this formula: **TP/(TP+FP)**
- To calculate **Recall** from confusion matrix we use this formula: **TP/(TP+FN)**
- The true positive rate and the false positive rate are combined in **ROC curves** to create an overall picture of classification performance.

Confusion Matrix for Training Data and Testing Data for Gradient Boosting model after Hyper parameter tuning



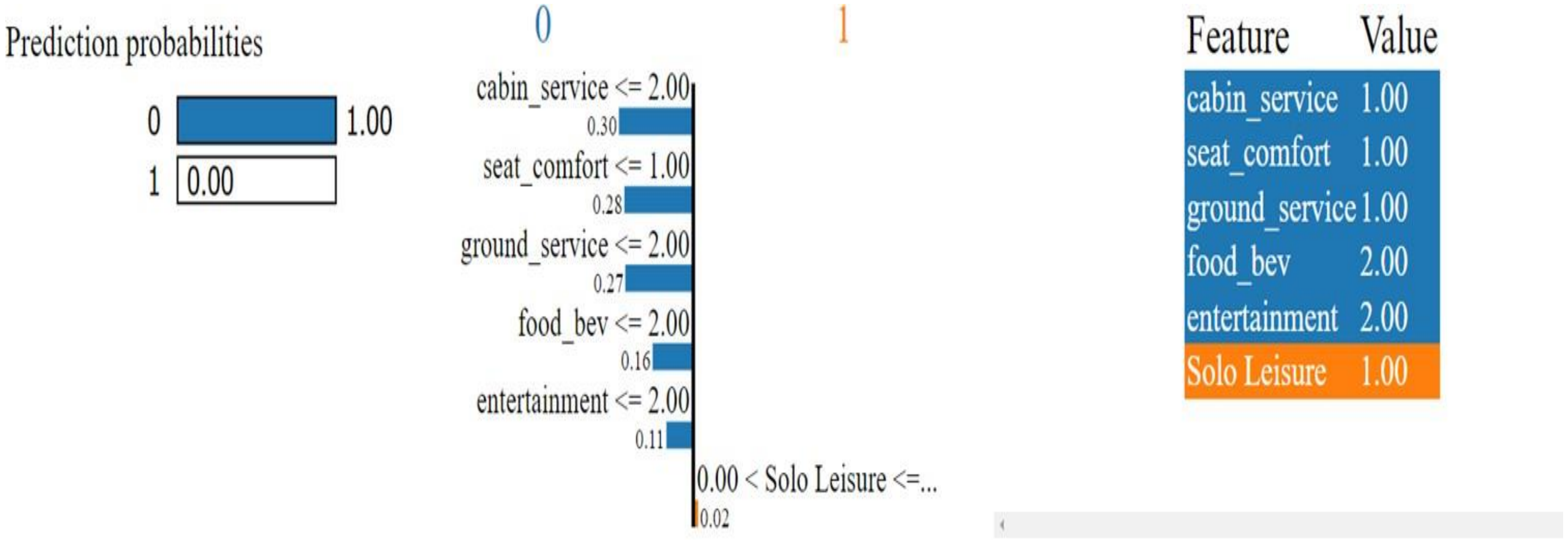
As in the above confusion matrix plot, we can see that for our testing data TP =5263, TN =6397, FP =489 and FN =739

Interpretability of Model:

- It's also crucial to interpret the model. The majority of machine learning models are blackbox-based.
- To understand what happens within the blackbox and why the model predicts certain outcomes.
- This is where **LIME** (Local Model Interpretability Model Agnostic) and **SHAP Values** comes in handy for interpretation.
- It is agnostic in nature.LIME Model interpret the results for single data points in any model. Whereas SHAP Model interprets for both single and entire datasets.
- We created an unseen data variable and stored our unseen data values of in that variable. Now this data is feed into LIME model for Gradient Boosting Model.
- For SHAP model, we feed entire training data to get interpretation of Gradient Boosting Model.

LIME Implementation:

```
#create a unseen data
unseen_data=np.array([1, 1, 2, 2, 1, 0, 0, 1, 1, 0, 0],dtype=float)
```



Feature	Value
cabin_service	1.00
seat_comfort	1.00
ground_service	1.00
food_bev	2.00
entertainment	2.00
Solo Leisure	1.00

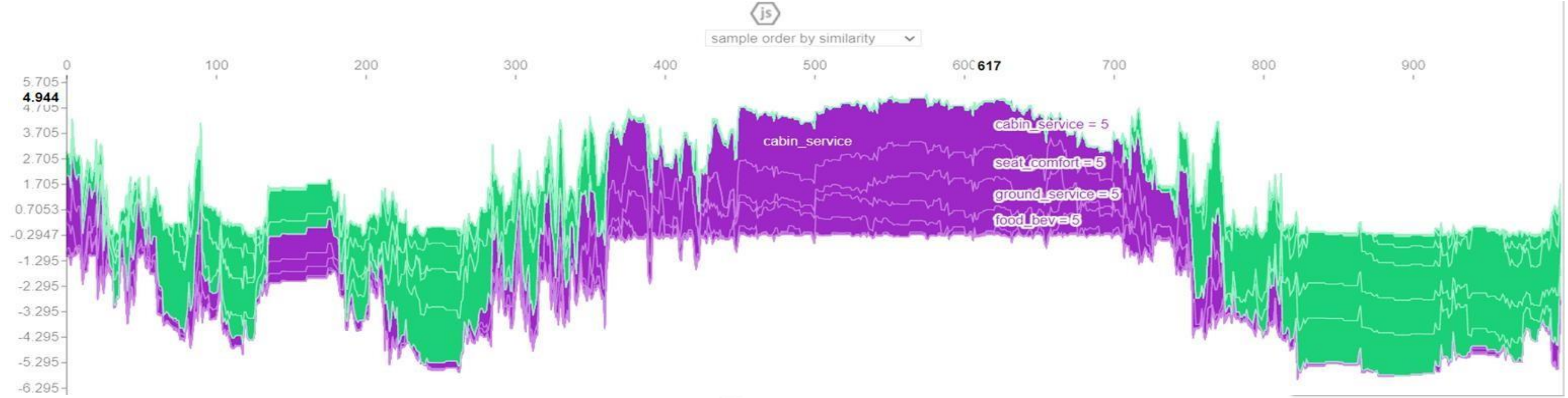
This plot Clearly says that as my Important Features like:

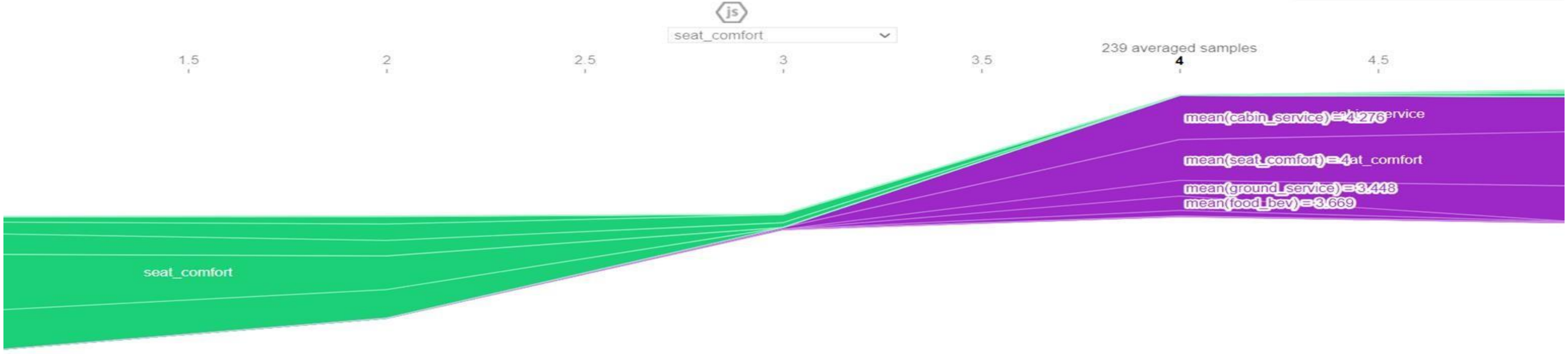
- **cabin_service=↓**
- **ground_service=↓**
- **seat_comfort=↓**

→ **Class '0' Labels** ↑

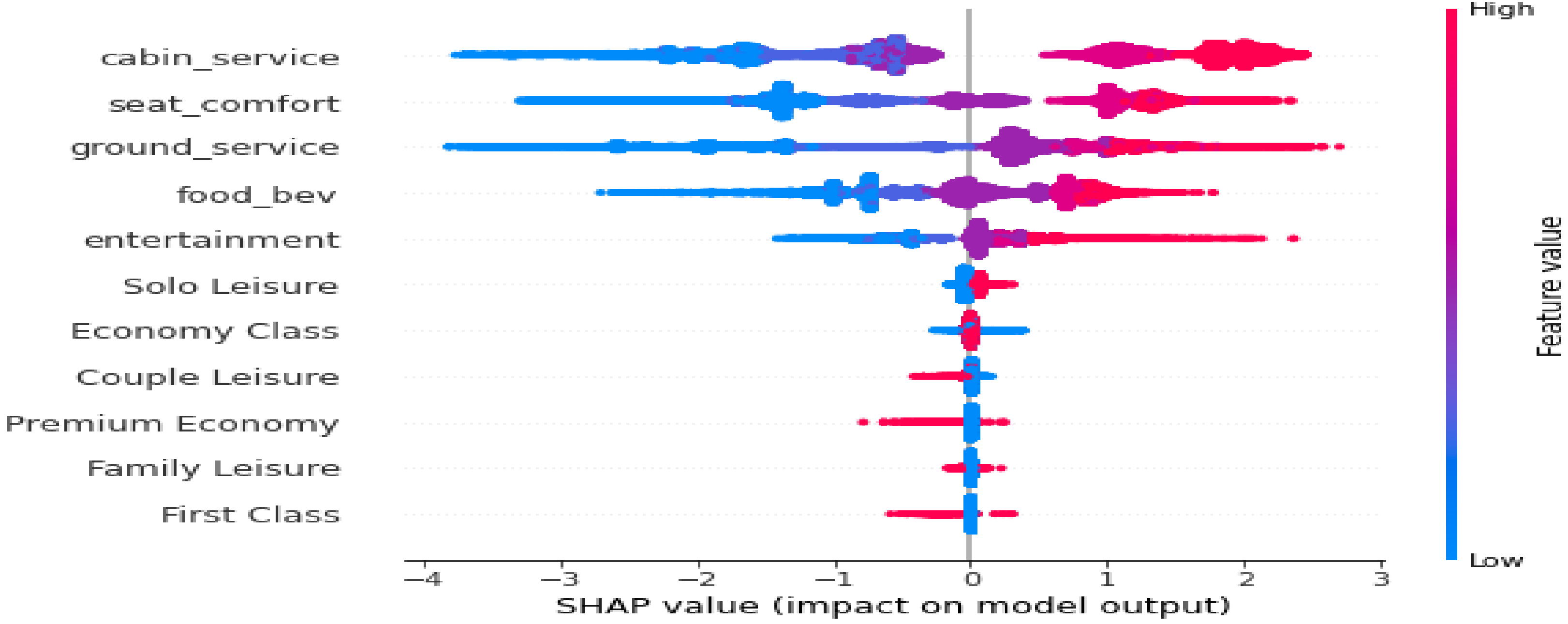
SHAP Implementation:

Force plot:





Summary plot:



From Summary plot it says that as my shap values of Important Features like:

- **cabin_service=↑**
- **seat_comfort=↑**
- **ground_service=↑**
- **entertainement=↑**

→ **Class '1' Labels** ↑

Conclusion:

- We started with our EDA and completed a variety of tasks. Using the EDA, we learn about the business understanding of our problem use case.
- We implemented four different models, such as Logistic Regression Model,Decision Tree Model, Random Forest Model, and Gradient Boosting Model.
- For Decision Tree Model, Random Forest Model, and Gradient Boosting Model, we used the Grid search CV method to do Hyperparameter tuning. This is done to increase accuracy and avoid Overfitting Criteria. After that, we finalised tuned the hyperparameters to the Gradient Boosting model.

- As a result of our knowledge of business and the problem use case, Recall is given top priority, Accuracy is given second priority, and ROC AUC is given third priority in the classification metrics. So that's why we finalised the tuned gradient boosting model.
- Now, the aviation industry may utilize this model to forecast passenger referral,examine the results, and make better decisions to improve their business needs and growth.
- Also they can provide additional benefits and rewards to passenger who made referrals.

Thank you