Introduction

The goal of the project is to use a flight status data set from 2018-2022 to build a machine learning model to predict whether a flight will suffer disruption before departure. According to the requirements, we can learn that this is a binary classification problem and it is supervised learning. In this project, the recall rate achieved at least 90% target, which means that our model can correctly predict 90% of disruption. Since I am using my own laptop, the selected data set is medium.

Data exploration and features selection

10 variables were selected in this project, with Disruption as the label value. Since this is a classification task, the chi-square test needs to be used to test whether each variable is related to Disruption. When p<0.05, the null hypothesis can be rejected and considered relevant. During the data exploration process, the selected attribute values include data categories such as object, int64 and float64, and there are 10 null values in the Disruption column. During data cleaning, records with null values need to be deleted and the float form converted to int form.

Features engineering

- Label Encoder converts airport names into numerical encodings, which can make the data easier to process by machine learning models.
- It is appropriate to use One-hot Encoding to convert the name of the airline into multiple binary feature columns because there is no clear sequential relationship between airlines and each airline should be treated as an independent feature in the model.
- Split the data set into training set, validation set and test set
  The original data set is divided into a training set (60%), a validation set (20%), and a test set (20%).

Training model

In the training model stage, I selected three models for calculation: random forest, decision tree and Gradient Boosting Classifier. The gradient boosting model achieved the highest recall, precision, and F1 scores, and achieved the largest area under the ROC curve. The precision-recall versus threshold curve shows that a threshold of approximately 0.1 must be chosen to achieve the desired recall score

Hyper parameter tuning

Random search is used here to tune the hyper parameters of the gradient boosting classifier. First, it is necessary to determine the hyper parameters to be optimized and their possible value ranges. By optimizing the hyper parameters, the model can be improved in the validation set or test set. Performance indicators, such as accuracy, recall, F1 score, etc. Secondly using random search allows for random sampling within a given hyper parameter space to find the best possible hyper parameter combination.

Testing and Conclusion

The adjusted model achieved the target of a prediction recall rate of 90%, that is, the model was able to effectively identify the majority of disrupted flights. Furthermore, it achieves an overall accuracy of approximately 28.98%. It has an accuracy of 21.45%, which means it is 21.00% accurate in predicting outages. In this practice, I found that the establishment of feature engineering has a relatively large impact on the results, including selecting more informative features, handling missing values and outliers, and encoding methods. If you need to further improve the model, you can further optimize the feature engineering process and explore new feature combination methods.