Introduction and Objectives

The goal of this project is to use a dataset of flights in the US between 2018 and 2022 to build a machine-learning model, capable of predicting future flight disruptions. The dataset provided contains various details about each flight, such as the date and time of the flight, the airports of departure and arrival, and the airline operating the flight. This is a supervised classification task, with the primary measure of performance being recall. The assumption is that passengers are less likely to be upset by a predicted disruption that never happens, compared to an unexpected flight cancellation. This project aims to achieve a recall score of at least 90%. This means that our model correctly predicts 90% of disruptions.

Data Exploration and Feature Selection

Since most of the features are categorical, correlation cannot be used as a selection criterion. Instead, a chi-square contingency test is conducted to determine if a feature is independent of a disruption. A p-value close to 0 suggested that the null hypothesis should be rejected, meaning that the two variables are related. Bar plots are also used to visualise the total number of flights, disrupted flights, and the ratio of flights to disrupted flights. The analysis of these metrics shows that the route of the flight, the airline operating the flight and the plane are good predictors of a disruption. Furthermore, there is a noticeable seasonal pattern in the frequency of disruptions with higher rates occurring in the summer months and late at night.

Feature Engineering and Preprocessing

Firstly, the data contains some missing values in the *Disruption* column. As this is the feature to be predicted, the corresponding columns are simply dropped from the data. The column $Tail\_Number$, which is a unique identifier for a plane, also contains missing values, all of which correspond to a disruption. To allow the model to learn the daily and annual seasonality, cyclical encoding is used to encode the features *Month* and $TimeOfDay$. This accurately captures distances between months such that for example January and December are equally far apart as January and February. For the features *Airline*, *Route*, and $Tail\_Number$ there is no sensible order to encode these categorical variables in and due to the large number of unique values it is infeasible to use a one-hot encoding. For this reason, target encoding is used as a method of encoding, which encodes each category as the mean of the target variable. The missing values are treated as another category such that the correspondence of a missing tail number and disruption is correctly captured. Features *Dest* and *Origin* are combined into a new feature *Route*, representing the route of the flight. Instead of cross-validation, a single validation set is used to allow the computer to deal with the computational load.

Model Selection

To select the most promising model for the given purpose, a Stochastic Gradient Descent, Random Forest and Histogram Gradient Boosting Classifier are compared where 0.5 is taken as a default threshold for a disruption. The latter is chosen for its ability to deal with large-scale datasets. For the Random Forest Classifiers the default parameters are manipulated to account for the amount of data the model is processing. The Random Forest model achieves the highest recall, accuracy and f1 score. The Gradient Boost achieves the highest area under the curve for the ROC curve. The Stochastic Gradient Descent Classifier has the highest precision but fails to predict many disruptions. Since the main goal is to identify as many positives as possible rather than correctly classifying each flight, the Random Forest model is chosen for further tuning. Furthermore, the precision-recall vs threshold curve suggests that a threshold of about 0.1 must be chosen to achieve the required recall score.

Hyperparameter tuning

Randomised searching is used for hyperparameter tuning to compare a set number of models. The models have parameters chosen from a specified distribution. This allows the comparison of a wider range of parameters without exhaustively checking all combinations. The cross-validation procedure is very memory expensive such that a sub sample from the data is collected. Stratified sampling is used to ensure that the sub sample maintains the same proportion of positives. The model is trained on the new parameters and validated with a threshold set to 0.1.

Testing and Conclusion

The tuned model meets the target of predicting more than 90% of disrupted flights. Furthermore, it achieves an overall accuracy of 31.00%, which is significantly higher than the accuracy of a naive model that always predicts a disruption (19.31%). It has a precision rate of 21.00%, which means it is accurate 21.00% of the time when predicting a disruption. This model could be improved with better computing hardware to allow further hyperparameter tuning or possibly data about the given weather conditions.