Machine Learning Project EDA

Explanatory Data Analysis Report

Team :
Simon Duchesne 11297912
Piero Matos 11339060
Balogog Georges 11337767

March 31, 2024

Introduction

- This presentation presents an exploratory analysis of flight information dataset.

Data Overview

Columns/Features

- FL_DATE, AIRLINE_CODE, DOT_CODE, FL_NUMBER, ORIGIN, ORIGIN_CITY, DEST, DEST_CITY, CRS_DEP_TIME, DEP_TIME, DEP_DELAY, TAXI_OUT, WHEELS_OFF, WHEELS_ON, TAXI_IN, CRS_ARR_TIME, ARR_TIME, ARR_DELAY, CANCELLED, CANCELLATION_CODE, DIVERTED, CRS_ELAPSED_TIME, ELAPSED_TIME, AIR_TIME, DISTANCE, DELAY_DUE_CARRIER, DELAY_DUE_WEATHER, DELAY_DUE_NAS, DELAY_DUE_SECURITY, DELAY_DUE_LATE_AIRCRAFT

Goal

We want to predict the *class* for a flight to **1)** delay its departure or arrival or not, based on a predefined threshold (some regulation define as critical a flight 3h late) and **2)** predict whether or not the flight is most likely to be cancelled in order to provide insights to customers to make informed decisions on a specific company to travel with.

Data Overview

Shape of Data

- 3,000,000 rows and 32 columns

Data Overview

Unique Values per columns

Column	Unique Values
FL_DATE	1704
AIRLINE	18
AIRLINE_DOT	18
AIRLINE_CODE	18
DOT_CODE	18
FL_NUMBER	7111
ORIGIN	380
ORIGIN_CITY	373
DEST	380
DEST_CITY	373
CRS_DEP_TIME	1384
DEP_TIME	1440
DEP_DELAY	1513
TAXI_OUT	179
WHEELS_OFF	1440
WHEELS_ON	1440
TAXI_IN	202
CRS_ARR_TIME	1435
ARR_TIME	1440
ARR_DELAY	1527
CANCELLED	2
CANCELLATION_CODE	4
DIVERTED	2
CRS_ELAPSED_TIME	640
ELAPSED_TIME	696
AIR_TIME	666
DISTANCE	1727
DELAY_DUE_CARRIER	1291
DELAY_DUE_WEATHER	812
DELAY_DUE_NAS	671
DELAY_DUE_SECURITY	172
DELAY_DUE_LATE_AIRCRAFT	958

Table: Unique Values per Column

Data Overview

NaN Values Proportion

Column	Nans	% Nans
FL_DATE	0.0	0.000000
AIRLINE	0.0	0.000000
AIRLINE_DOT	0.0	0.000000
AIRLINE_CODE	0.0	0.000000
DOT_CODE	0.0	0.000000
FL_NUMBER	0.0	0.000000
ORIGIN	0.0	0.000000
ORIGIN_CITY	0.0	0.000000
DEST	0.0	0.000000
DEST_CITY	0.0	0.000000
CRS_DEP_TIME	0.0	0.000000
DEP_TIME	77615.0	2.587167
DEP_DELAY	77644.0	2.588133
TAXI_OUT	78806.0	2.626867
WHEELS_OFF	78806.0	2.626867
WHEELS_ON	79944.0	2.664800
TAXI_IN	79944.0	2.664800
CRS_ARR_TIME	0.0	0.000000
ARR_TIME	79942.0	2.664733
ARR_DELAY	86198.0	2.873267
CANCELLED	0.0	0.000000
CANCELLATION_CODE	2920860.0	97.362000
DIVERTED	0.0	0.000000
CRS_ELAPSED_TIME	14.0	0.000467
ELAPSED_TIME	86198.0	2.873267
AIR_TIME	86198.0	2.873267
DISTANCE	0.0	0.000000
DELAY_DUE_CARRIER	2466137.0	82.204567
DELAY_DUE_WEATHER	2466137.0	82.204567
DELAY_DUE_NAS	2466137.0	82.204567
DELAY_DUE_SECURITY	2466137.0	82.204567
DELAY_DUE_LATE_AIRCRAFT	2466137.0	82.204567

Table: NaN Values Proportion

Data Overview

Delay Time

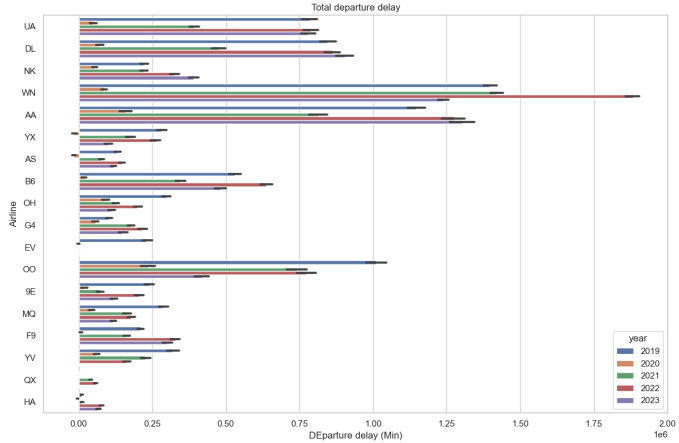

Figure: Total departure delay

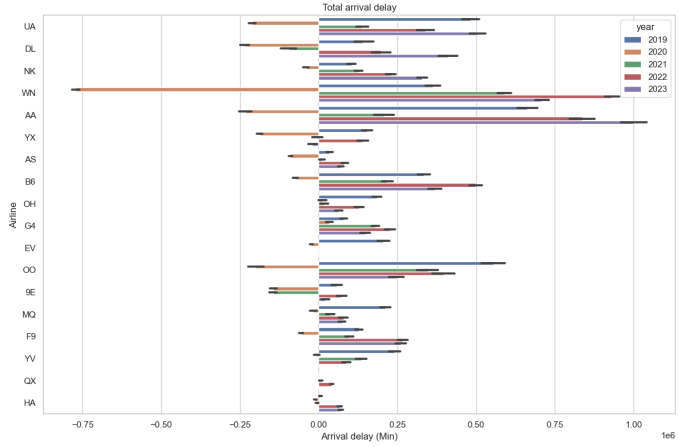

Figure: Total arrivall delay

Cancellation Analysis

Number of Occurrences of Different Classes

- Class 0 (Not Cancelled): 2,920,860 occurrences (97.362%)
- Class 1 (Cancelled): 79,140 occurrences (2.638%)

Cancellation Analysis

Figures

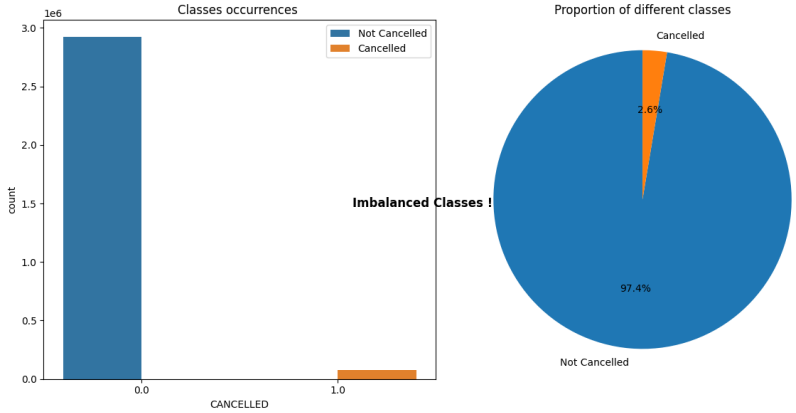

Figure: Classes proportion

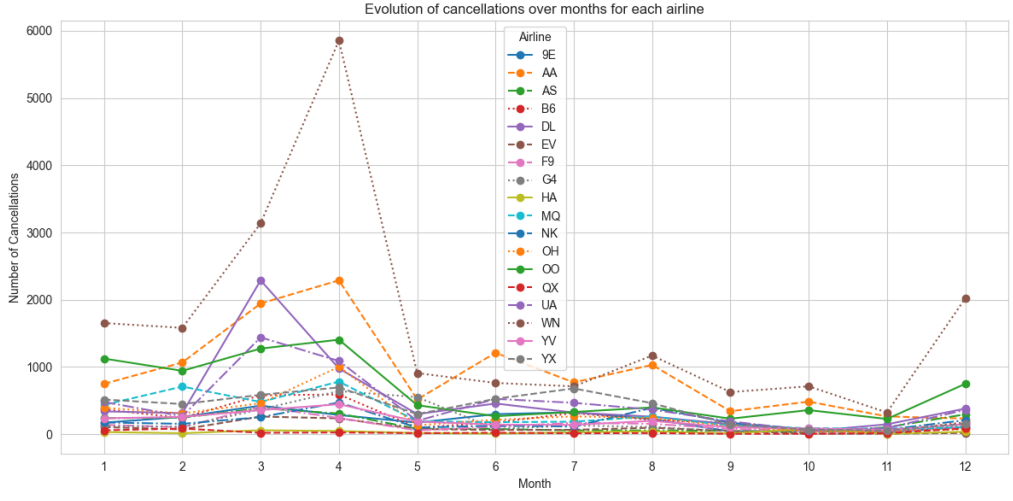

Figure: Evolution of cancellations over months for each airline

Cancellation Analysis

Figures

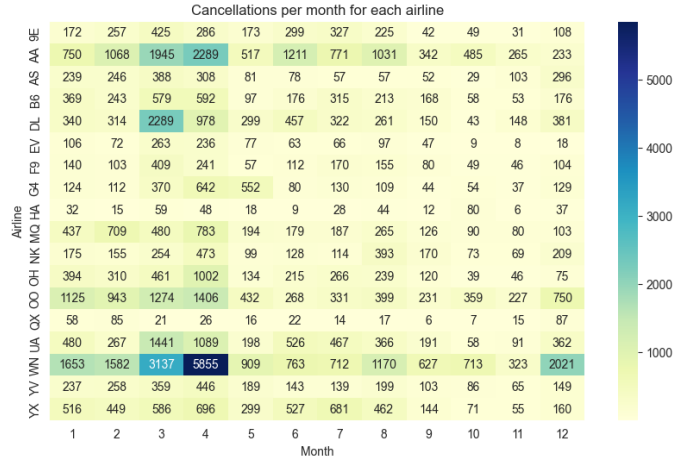

Figure: Cancellations per month for each airline

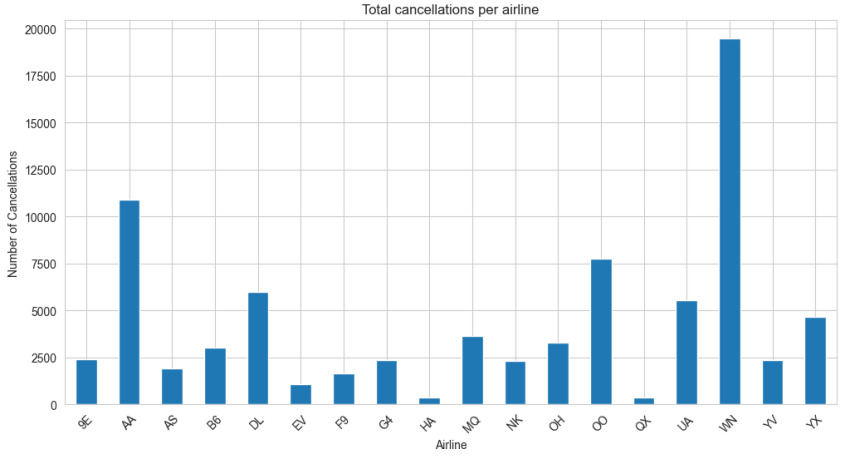

Figure: Total cancellations per airline

Preprocessing technics proposal

Pre-processing

- One hot encoding (OHE) of features
  - **Origin_city**
  - **Destination_city**
  - **Airline_code**
- Discritizing the delay feature (arrival)
  - By discretizing the delay in oder to obtain multiple classes, we can predict how critical the flight is. We can use **KbinsDiscritizer** or just **map** using mapping function then OHE.
- Distance normalization
- Creating new feature : **ARR TYPE HOUR CRS** : The schedule arrival time
  ['Morning 0-12', 'Afternoon 12-17', 'Night 17-24']

Features and Outputs selection

Features

- Features:
  - **AIRLINE CODE** (OHE)
  - **ORIGIN** (OHE)
  - **DEST** (OHE)
  - **DISTANCE**
  - **ARR TYPE HOUR CRS** (OHE)
- Output :
  - **ARR TYPE DELAY** :
    bins = [-np.inf, -0.9999, 3 , 50 , np.inf] (in hours)
    labels = ['Early', 'Very Good', 'Bad', 'Cancelled']

Baseline

SCV - Support vector classifier

We're gonna use an SVM model fit with the **ovr** (one vs rest) approach as baseline model; usig the **scikit learn** library :

'from sklearn.svm import SVC'

- Score: It is computed on the **macro** approach
  'f1-score': 0.32,
  'recall-score': 0.33,
  'accuracy-score': 0.96,
  'precision-score': 0.32

Conclusion

- This explanatory analysis provides an overview of the dataset, including column details, data shape, unique values, missing values proportion, statistics, and cancellation analysis.