

**AIR SERVICE DELAY FORETELL USING MACHINE
LEARNING**

at

**Sathyabama Institute of Science and Technology
(Deemed to be University)**

Submitted in partial fulfillment of the requirements for the award
of Bachelor of Technology Degree in
Information Technology

By

RAMYA U

(Reg. No. 37120057)

VARSHA M

(Reg. No. 37120081)



**DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI-600119
MARCH - 2021**



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" Grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600 119.

Phone: 044 - 2450 3150 / 51 / 52 / 54 / 55 Fax: 044 - 2450 2344

www.sathyabama.ac.in



SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of RAMYA U (37120057), VARSHA M (37120081) who carried out the project entitled "**AIR SERVICE DELAY FORETELL USING MACHINE LEARNING**" under my supervision from November 2019 to April 2020.

Internal Guide

Dr. SENDURU SRINIVASULU, MTech., Ph.D.,

Head of the Department

Dr. R. SUBHASHINI, M.E., Ph.D.,

Submitted for Viva voce Examination held

on _____

Internal Examiner

External Examiner

DECLARATION

We, RAMYA U (37120057), VARSHA M (37120081) hereby declare that the Project Report on entitled **“AIR SERVICE DELAY FORETELL USING MACHINE LEARNING”** done by me under the guidance of Dr. SENDURUSRINIVASULU., at Sathyabama Institute Of Science And Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of technology degree in Information Technology.

DATE:

SIGNATURE OF THE CANDIDATE

PLACE:

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them

I convey my thanks to Dr. T.Sasikala M.E., Ph.D, Dean,School of Computing , Dr.SUBHASHINI M.E., Ph.D. , Head of the Department of Information and Technology for providing me necessary support and details at the right time during the progressive reviews.

I would like to express y sincere and deep sense of gratitude to my Project Guide Dr.SENDURUSRINIVASULU., for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Information Technology who were helpful in many ways for the completion of the project.

ABSTRACT

AIR SERVICE DELAY FORETELL USING MACHINE LEARNING

Passenger all over the world travel, at their needs in order to fly over to their required destination in absolutely hassle-free time for their vacation business purpose or etc, but due to the current corruption being occurred in the air traffic control mean mode of transportation there has been a huge negativity being occurred of causing delay in the flights which has made people more worried in not being able to reach their destination on time. In order to give an pre-assumption of predicting on what basis these flight are becoming delayed we create an model and well train it in order to calculate the problem occurred from an particular region with an estimation of few international airports and some international airlines we are preparing an multiagent which can detect and solve the issue giving us an clear image of the statistics of the atmosphere present in those particular airports we train and test the model to calculate and predict few of the airports for a period of a month or two and classify them into delayed aircrafts and regular aircrafts on time flights with a result we could bring down an percentage 70% with an result of delay aircrafts within the time period of 20 minutes to bring to notice and more elaborated scheme we have evaluated the number of ratio total days delays caused in an aircraft to our measure we could say that major of the delay were ranging between 5% and that to they are basic delays which have being occurred therefore our method can classify and bring up an accurate scheduling of flights with higher classification methods of prediction.

TABLE OF CONTENTS

<u>CHAPTER NO.</u>	<u>TITLE</u>	<u>PAGE NO.</u>
	ABSTRACT	v
	LIST OF FIGURES	viii
	LIST OF ABBREVIATIONS	ix
1.	INTRODUCTION	9
	1.1 PROJECT OUTLINE	
	1.2 INTRODUCTION	
2.	LITERATURE SURVEY	11
3.	METHODOLOGY	15
	3.1 DATASET	
	3.2 DATA PRE-PROCESSING	
	3.3 DATA PROCESSING	
	3.4 CLASSIFICATION MODEL	
	3.5 PREDICTION OF DELAYS USING REGRESSION	
4.	RESULTS AND DISCUSSION	18
5.	CONCLUSION AND FUTURE WORK	19
	REFERENCE	20
	APPENDIX	21
	A. SOURCE CODE	
	B. SCREEN SHOTS	

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
1.2.1	Architecture diagram	3
3.2.1	Data Pre-processing	8
3.4.1	Classification Model	10
4.1.1	prediction graph	11
4.1.2	prediction values	11

LIST OF ABBREVIATION

IATA	-	INTERNATIONAL AIR TRANSPORT ASSOCIATION
RMSE	-	THE ROOT MEAN SQUARE ERROR
SMOTE	-	SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE
NASA	-	NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

CHAPTER 1

INTRODUCTION

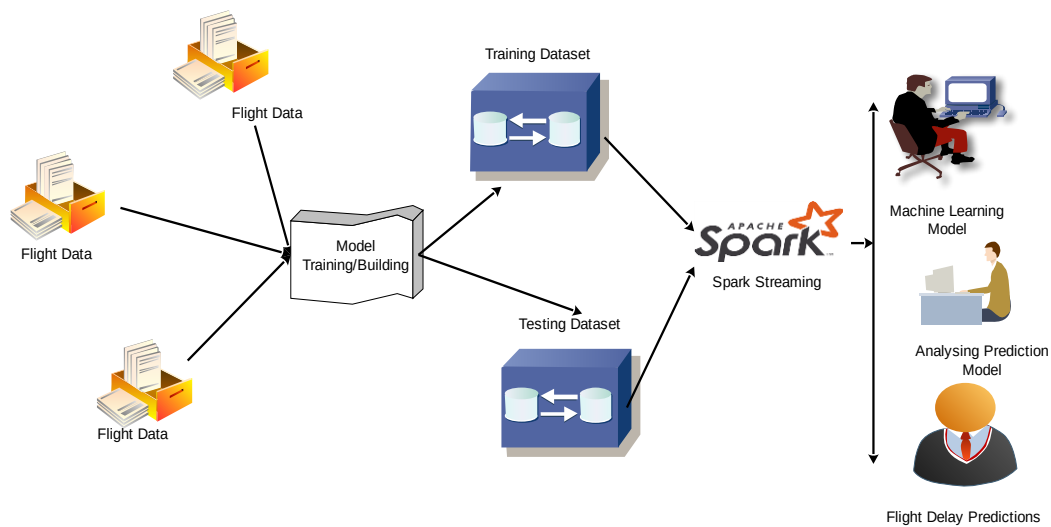
1.1 PROJECT OUTLINE:

TITLE OF THE PROJECT: - AIR SERVICE DELAY FORETELL MACHINE LEARNING

1.2 INTRODUCTION:

The international aviation is a huge connection dealing with numerous number of flights and to ensure the safety of the passenger flying through aircrafts and providing an scheduled transport of passengers, the department has an result in a massive number of coverages in all means of flight transport the IATA team predicts 10.2 billion passengers by the year of 2040 would be travelling at an massive growth rate of the population with the massive increase in air traffic control system we have been highly checking and enhancing to develop the information from the airports and the reason of the flight delays being occurred and caused each and every day in an particular city airport it has been revealed that most of the flight delays have been caused in the ground region the delay because of an particular airport leading to disturbed environment in many of the neighbouring airports causing delays the last few years the main reason for such delays have been massive through calculation of the statistics and some methods and approaches using machine learning and multiagent based solution we could create an model through which it will calculate and tell us the exact reason of the fall in delay propagations thanks to the theory of one main scientist he had developed an approach to record and provide the information of whether the delay have been occurred in air or at ground due to any faults in the staffs handling their calculation states that in 50 minutes of time they have initialized an average of 70% of delays have been caused at the ground staff issues , with the help of one more scientist we were very easily being able to detect the delay in any chance if it had been caused in the air due to any weather conditions and to our results we could tell that 30% of delays have been also formed because of the weather conditions and an scientist named 'CHOI ET AL' well trained an model to predict

The cause or delay using many algorithm and methods such as decision tree and random forest classifier to well understand the reason of the delays. the latest researchers have brought up with an exclusive idea to change their prediction way of estimating from machine learning models and bring up and change from the feature selection process which have basically resulted in errors in the department of air traffic multiagent based performance also resulted in giving passengers an clear show of the next flight will be arrived or departed on what time in an accurate manner few days ago agent-based methods have been used in determining flight scheduling and mostly the safety of many passengers as well as pilots who risk their life in order to transfer people at their required destination an well famous scientist had also developed an source through which they check the parameters of one way flights and multiple round way connection flights and disturbance.



1.2.1 Architecture diagram

CHAPTER 2

LITERATURE SURVEY

TITLE: Unsupervised Flight Phase Recognition with Flight Data Clustering based on GMM

AUTHOR: Datong Liu, Ning Xiao, Xiyuan Peng

YEAR: 2020

DESCRIPTION:

Currently, with the rapid development of the aviation industry, researchers are paying more attention to the improvement of aviation safety. Aviation safety mainly includes flight safety, aviation ground safety, and air defense safety. In terms of flight safety, the analysis of large amounts of flight data has gradually become a useful tool for timely detection of potential dangers at various stages of flight. As a result, flight data analysis has been one of the hot topics in aviation. However, due to the complexity of the aircraft operating conditions throughout the aircraft, if the data is analyzed at the entire flight phase, it is very difficult and time consuming to identify the problematic flight phase. Therefore, flight phase recognition for civil aircraft is implemented in this study. A flight phase recognition method based on Gaussian Mixture Model (GMM) is proposed in this work, which is the important foundation for timely detecting the abnormal event and improving the system safety and reliability. Firstly, the FDR data are preprocessed by spline interpolation and normalization, and then a GMM-based flight phase clustering is realized. In addition, a set of evaluation method is developed to evaluate the quality of flight phase recognition result. Finally, the effectiveness of the method is verified by using real FDR data from NASA's open database.

TITLE: Software Solution for Visualization and Evaluation of Flight Data in Terms of Competency-Based Training

AUTHOR: Vladimír Socha, Roman Matyáš, Stanislav Kušmírek

YEAR: 2020

DESCRIPTION:

In this paper, we present a complex solution for flight data gathering, analyzation, and visualization. This solution can be used for the Competency-Based Training of pilots in general aviation. For the data gathering, we are introducing a concept of an embedded device. For data analyzation, we are introducing the approach for algorithmic detection of flight maneuvers based on the recorded flight data. There is a multiplatform software application for data visualization. We are also describing the reasons for creating such solutions and potential future uses of our hardware and software products. This paper is mostly describing the important requirements and their implementation regarding the future usage for Competency-Based Training of pilots in general aviation. We are not describing a lot of technical details to keep this paper straightforward and we are mostly focusing on passing the whole idea and benefits of our products.

TITLE: Development of Autonomous UAV Quadcopters using Pixhawk Controller and Its Flight Data Acquisition

AUTHOR: Karlisa Priandana, Muhammad Hazim, Wulandari

YEAR: 2020

DESCRIPTION:

Unmanned Aerial Vehicle (UAV) quadcopter is a copter with four driving motors. It is widely used in various applications including agriculture because the design is simple and the concept of motion is easy to understand. However, currently, the widely sold quadcopters in the Indonesian market are quadcopters that are controlled manually using remote control. In this work, we developed two autonomous UAV quadcopters that can fly autonomously according to two predetermined trajectories, i.e., rectangular and circular trajectories. This preliminary work aims to support various quadcopter applications for smart and precision farming on fix agricultural lands. The trajectories are predetermined in accordance to the landforms and the quadcopters are expected to follow these trajectories autonomously. This study has succeeded in building two autonomous quadcopter with different specifications: quadcopter type F450 and quadcopter type F330. The experimental testing results show that the average latitude and longitude mean squared error of the quadcopter flight is 6.79×10^{-10} for the F450 quadcopter and 1.45×10^{-8} for the F330 quadcopter. The flight sensors data obtained in this study is available to be used for research and educational purposes. Please contact the authors for access to the UAV flight data.

TITLE: Predicting Flight Delays with Error Calculation using Machine Learned Classifiers

AUTHOR :Priyanka Meel, Mukul Singha I, Mukul Tanwar

YEAR: 2020

DESCRIPTION:

Flight delay is a major problem in the aviation sector. During the last two decades, the growth of the aviation sector has caused air traffic congestion, which has caused flight delays. Flight delays result not only in the loss of fortune also negatively impact the environment. Flight delays also cause significant losses for airlines operating commercial flights. Therefore, they do everything possible in the prevention or avoidance of delays and cancellations of flights by taking some measures. In this paper, using machine learning models such as Logistic Regression, Decision Tree Regression, Bayesian Ridge, Random Forest Regression and Gradient Boosting Regression we predict whether the arrival of a particular flight will be delayed or not.

CHAPTER 3

METHODOLOGY

3.1 DATASET:

We collect our own requirements which are needed in order to calculate the delay which have being occurred in the airport some of the major requirements are like collecting information on the routes of each airlines from where they have been traveling and their departure location and their arrival location thought which we will be able to estimate the reason and problem occurred in such cases so these valuable information which we have collected are being kept to pursue further with throughout our journey in estimating mainly even the connecting aircrafts details also are being taken under our record.

3.2 DATA PRE-PROCESSING:

This method is basically initialised before applying our algorithm in order to correct the format of our calculation we use this data pre-processing which will enable us to detect the right way to imply in our model and then we will be able to apply our algorithms in an proper way in many sources we could find that real time data has not being implemented correctly , sample data set from the ministry of aviation department and we could find that there were so many missing forms of values the data's were changed with the help of pandas () function , through our method of initialising the data pre-processing we could clarify the rows and columns correctly and initialised them in an sequence manner.



```
Python console
Console 1/1A
After preprocessing
YEAR 0
MONTH 0
DAY 0
DAY_OF_WEEK 0
AIRLINE 0
FLIGHT_NUMBER 0
TAIL_NUMBER 0
ORIGIN_AIRPORT 0
DESTINATION_AIRPORT 0
SCHEDULED_DEPARTURE 0
DEPARTURE_TIME 0
DEPARTURE_DELAY 0
TAXI_OUT 0
WHEELS_OFF 0
SCHEDULED_TIME 0
ELAPSED_TIME 0
AIR_TIME 0
DISTANCE 0
WHEELS_ON 0
TAXI_IN 0
SCHEDULED_ARRIVAL 0
ARRIVAL_TIME 0
ARRIVAL_DELAY 0
DIVERTED 0
CANCELLED 0
dtype: int64
(54885, 25)
```

Fig 3.2.1 Data Pre-processing

3.3 DATA PROCESSING:

After collecting many dataset from various flight department we came to find out that there were many instances nearly 100000 details of flights which had almost many missing details or dates of them so we were well enough to remove them from our process and go ahead with the instances which had all the major requirements in handy which would make the work for also most speedily to complete in an hassle free time with the rest requirements such as the place of arrival , place of departure throughout where it has been traveling either from an single route or an connection route all the required details of which (month,day,date,place,) all these were included and more over we could even calculate the airport name and their code we would remove most of such parameters mostly which were not required in our detection or prediction of the flight delay prediction

3.4 CLASSIFICATION MODEL:

In this step we clearly tell you that we use main algorithm such as (I)Gradient boosting, (ii)Knn, (iii)Random forest classifier, with the help of these algorithm we could classify and bring down the instances from major of the fast moving major airports in the world and clarify the details such as the origin of the flight from which exact location has it been take up and through which place it would like to travel these method have been clearly implemented and gave us an easy go for training our model with the help of the dataset which we have already collected we were able to easy train our model in classifying the delay cause and reason due to these negativity there was also an rule of the bts being included through which we would collect the time from between the flight to reach at its estimated gate at what times and speed it would travel and the delay for it to take to reach not in the correct time these most points have been given to our training set to run and find out the problem an scaling and featuring method has also being linked with our algorithm in order to enhanced and execute our dataset with an matrix included form which will release the accuracy of our system and provide us with the exact role and prediction our delays and hence through these information which have given to us we were able to retrieve that data.

Supervised Machine Learning: Classification & Regression

Classification



Identifies category for item

Fig 3.4.1 Classification Model

3.5 PREDICTION OF DELAYS USING REGRESSION:

It is been proved that from various notices that between the origin airports and all the airlines, that there is a high difference between in average delays, both noticed between different airports but also between different airlines. This is different cause it implies that in order to correctly model the delays, it is necessary to bring up a model that is exact to the company and the home airport. After the discovery of dataset, the final form was to achieve is to bring up a model for prediction of delays. The prediction is retrieved using a three-week calculation form through that we will predict the delays for the following week this process keeps flowing and we retrieve our predictions on time.

CHAPTER 4

RESULTS AND DISCUSSION

This machine learning model was able to predict the arrival delay of a flight with a decent accuracy. Boost Classifier was able to classify the data into delayed or non-delayed flights, and Boost Regressor was able to predict the arrival delay in minutes. The Boost classifier was able to give an F1 score of 0.93 for the majority class and 0.75 for the minority class after SMOTE oversampling. The predicted arrival delay had a Mean Absolute Error of close to 11 minutes with the actual arrival delay. The RMSE was close to 16 minutes with the actual arrival delay.



Fig 4.1.1 prediction graph



Fig 4.1.2 prediction values

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 CONCLUSION:

Major part of the world people really face many problem at the time they are travelling especially when it could be an important travel time they get delayed because of the delay occurred in the flights to bring down this problem we were able to transcript and make an model which could relate and tell us the issue or the reason in occurrence of such type of faults which have been taken place our plan brings up an multiagent based method which helps us in classify the delays each and every flights have an particular agent through which they can tell us about the environment of the status of each and every flight that they have being travelled with this complex connection of major number of agents with the airport atmosphere we can schedule the accurate flow of the primary delay of the flights if any have being occurred in the upcoming or further delay if they have being occurred.

5.2 FUTURE WORK:

Future work we like to further improve our models, perhaps with more training-data or deeper neural network, or both. Taxi-delay prediction is a natural progression to this work, considering amount of fuel wasted while taxiing. Accurate taxi-delay prediction requires taking airport runway and taxiway configurations in to consideration where very little work exists.

REFERENCES

- [1] Michael Ball, Cynthia Barnhart, Martin Dresner, Mark Hansen, Kevin Neels, Amedeo Odoni, Everett Peterson, Lance Sherry, Antonio Trani, Bo Zou (2010). 'Total Delay Impact Study', The National Center of Excellence for Aviation Operation Research (NEXTOR)
- [2] (Online) Bureau of Transportation Statistics (BTS) Databases and Statistics.
<http://www.transtats.bts.gov/>
- [3] Nicholas G. Rupp (2007). 'Further Investigations into the Causes of Flight Delays', Department of Economics, East Carolina University.
- [4] Juan Jose Rebollo and Hamsa Balakrishnan (2014). 'Characterization and Prediction of Air Traffic Delays', Massachusetts Institute of Technology
- .
- [5] Diminique Burgauer and Jacob Peters (2000). 'Airline Flight Delays and Flight Schedule Padding', University Of Pennsylvania.
- [6] Q. L. Qin and H. Yu (2014), A statistical analysis on the periodicity of flight delay rate of the airports in the US, Advances in Transportation Studies an international Journal 2014 Special Issue, Vol. 3
- .
- [7] Eric R. Mueller and Gano B. Chatterji. 'Analysis of Aircraft arrival and departure delay characteristics', NASA Ames Research Center, Moffett Field, CA 94035-1000.
- [8] Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani (2013), An Introduction to statistical Learning with applications in R, Springer.
- [9] Breiman, L. Random Forests. Mach. Learn. 2001, 45, 5-32.
- [10] Breiman, L.; Friedman, J. H.; Olshen, R. A.; Stone, C. J. Classification and Regression Trees; Chapman & Hall/CRC: Boca Raton, 1984.

APPENDIX

A. SOURCE CODE

```
import datetime, warnings, scipy
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib as mpl
import matplotlib.pyplot as plt
import matplotlib.patches as patches
from matplotlib.patches import ConnectionPatch
from collections import OrderedDict
from matplotlib.gridspec import GridSpec
from sklearn import metrics, linear_model
from sklearn.preprocessing import PolynomialFeatures, StandardScaler
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
from sklearn.model_selection import train_test_split, cross_val_score,
cross_val_predict
from scipy.optimize import curve_fit
plt.rcParams["patch.force_edgecolor"] = True
plt.style.use('fivethirtyeight')
mpl.rcParams['patch', edgecolor = 'dimgray', linewidth=1)
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "last_expr"
pd.options.display.max_columns = 50
warnings.filterwarnings("ignore")

df = pd.read_csv('C:\\Users\\ELCOT\\Desktop\\FD\\flights.csv',
low_memory=False)
print('Dataframe dimensions:', df.shape)
#
# gives some infos on columns types and number of null values
tab_info=pd.DataFrame(df.dtypes).T.rename(index={0:'column type'})
tab_info=tab_info.append(pd.DataFrame(df.isnull().sum()).T.rename(index={0:
'null values (nb)'}))
tab_info=tab_info.append(pd.DataFrame(df.isnull().sum()/df.shape[0]*100)
.T.rename(index={0:'null values (%)'}))
tab_info

airports = pd.read_csv("c:\\users\\ELCOT\\Desktop\\FD\\airports.csv")
```

```

plt.figure(figsize=(11,11))
#
# define properties of markers and labels
colors = ['yellow', 'red', 'lightblue', 'purple', 'green', 'orange']
size_limits = [1, 100, 1000, 10000, 100000, 1000000]
labels = []
for i in range(len(size_limits)-1):
    labels.append("{} <.< {}".format(size_limits[i], size_limits[i+1]))
#
map = Basemap(resolution='i',llcrnrlon=-180, urcrnrlon=-50,
              llcrnrlat=10, urcrnrlat=75, lat_0=0, lon_0=0,)
map.shadedrelief
map.drawcoastlines()
map.drawcountries(linewidth = 3)
map.drawstates(color='0.3')
#
# put airports on map
for index, (code, y,x) in airports[['IATA_CODE', 'LATITUDE',
'LONGITUDE']].iterrows():
    x, y = map(x, y)
    isize = [i for i, val in enumerate(size_limits) if val < count_flights[code]]
    ind = isize[-1]
    map.plot(x, y, marker='o', markersize = ind+5, markeredgewidth = 1, color =
colors[ind],
            markeredgewidth=1, label = labels[ind])
#
# remove duplicate labels and set their order
handles, labels = plt.gca().get_legend_handles_labels()
by_label = OrderedDict(zip(labels, handles))
key_order = ('1 <.< 100', '100 <.< 1000', '1000 <.< 10000',
            '10000 <.< 100000', '100000 <.< 1000000')
new_label = OrderedDict()
for key in key_order:
    new_label[key] = by_label[key]
plt.legend(new_label.values(), new_label.keys(), loc = 1, prop= {'size':11},
          title='Number of flights per year', frameon = True, framealpha = 1)
plt.show()

df = df[df['MONTH'] == 1]

df['DATE'] = pd.to_datetime(df[['YEAR', 'MONTH', 'DAY']])

```

```

#
# Function that convert the 'HHMM' string to datetime.time
def format_heure(chaine):
    if pd.isnull(chaine):
        return np.nan
    else:
        if chaine == 2400: chaine = 0
        chaine = "{0:04d}".format(int(chaine))
        heure = datetime.time(int(chaine[0:2]), int(chaine[2:4]))
        return heure
#

# Function that combines a date and time to produce a datetime.datetime
def combine_date_heure(x):
    if pd.isnull(x[0]) or pd.isnull(x[1]):
        return np.nan
    else:
        return datetime.datetime.combine(x[0],x[1])
#

# Function that combine two columns of the dataframe to create a datetime
format
def create_flight_time(df, col):
    liste = []
    for index, cols in df[['DATE', col]].iterrows():
        if pd.isnull(cols[1]):
            liste.append(np.nan)
        elif float(cols[1]) == 2400:
            cols[0] += datetime.timedelta(days=1)
            cols[1] = datetime.time(0,0)
            liste.append(combine_date_heure(cols))
        else:
            cols[1] = format_heure(cols[1])
            liste.append(combine_date_heure(cols))
    return pd.Series(liste)

df['SCHEDULED_DEPARTURE'] = create_flight_time(df,
'SCHEDULED_DEPARTURE')
df['DEPARTURE_TIME'] = df['DEPARTURE_TIME'].apply(format_heure)
df['SCHEDULED_ARRIVAL'] =
df['SCHEDULED_ARRIVAL'].apply(format_heure)
df['ARRIVAL_TIME'] = df['ARRIVAL_TIME'].apply(format_heure)
#

```

```

variables_to_remove = ['TAXI_OUT', 'TAXI_IN', 'WHEELS_ON',
                        'WHEELS_OFF', 'YEAR',
                        'MONTH', 'DAY', 'DAY_OF_WEEK', 'DATE',
                        'AIR_SYSTEM_DELAY',
                        'SECURITY_DELAY', 'AIRLINE_DELAY',
                        'LATE_AIRCRAFT_DELAY',
                        'WEATHER_DELAY', 'DIVERTED', 'CANCELLED',
                        'CANCELLATION_REASON',
                        'FLIGHT_NUMBER', 'TAIL_NUMBER', 'AIR_TIME']
df.drop(variables_to_remove, axis = 1, inplace = True)
df = df[['AIRLINE', 'ORIGIN_AIRPORT', 'DESTINATION_AIRPORT',
        'SCHEDULED_DEPARTURE', 'DEPARTURE_TIME',
        'DEPARTURE_DELAY',
        'SCHEDULED_ARRIVAL', 'ARRIVAL_TIME', 'ARRIVAL_DELAY',
        'SCHEDULED_TIME', 'ELAPSED_TIME']]
df[:5]

missing_df = df.isnull().sum(axis=0).reset_index()
missing_df.columns = ['variable', 'missing values']
missing_df['filling factor (%)']=(df.shape[0]-missing_df['missing
values'])/df.shape[0]*100
missing_df.sort_values('filling factor (%)').reset_index(drop = True)

df.dropna(inplace = True)

airlines_names = pd.read_csv('C:\\Users\\ELCOT\\Desktop\\FD\\airlines.csv')
airlines_names

abbr_companies =
airlines_names.set_index('IATA_CODE')['AIRLINE'].to_dict()
# function that extract statistical parameters from a grouby objet:
def get_stats(group):
    return {'min': group.min(), 'max': group.max(),
            'count': group.count(), 'mean': group.mean()}
# Creation of a dataframe with statitcal infos on each airline:
global_stats =
df['DEPARTURE_DELAY'].groupby(df['AIRLINE']).apply(get_stats).unstack()
global_stats = global_stats.sort_values('count')
global_stats

```

```

font = {'family' : 'normal', 'weight' : 'bold', 'size' : 15}
mpl.rc('font', **font)
import matplotlib.patches as mpatches
#


---


# I extract a subset of columns and redefine the airlines labeling
df2 = df.loc[:, ['AIRLINE', 'DEPARTURE_DELAY']]
df2['AIRLINE'] = df2['AIRLINE'].replace(abbr_companies)
#


---


colors = ['royalblue', 'grey', 'wheat', 'c', 'firebrick', 'seagreen', 'lightskyblue',
          'lightcoral', 'yellowgreen', 'gold', 'tomato', 'violet', 'aquamarine',
          'chartreuse']
#


---


fig = plt.figure(1, figsize=(16,15))
gs=GridSpec(2,2)
ax1=fig.add_subplot(gs[0,0])
ax2=fig.add_subplot(gs[0,1])
ax3=fig.add_subplot(gs[1,:])
#-----
# Pie chart n°1: nb of flights
#-----
labels = [s for s in global_stats.index]
sizes = global_stats['count'].values
explode = [0.3 if sizes[i] < 20000 else 0.0 for i in range(len(abbr_companies))]
patches, texts, autotexts = ax1.pie(sizes, explode = explode,
                                   labels=labels, colors = colors, autopct='%1.0f%%',
                                   shadow=False, startangle=0)
for i in range(len(abbr_companies)):
    texts[i].set_fontsize(14)
ax1.axis('equal')
ax1.set_title('% of flights per company', bbox={'facecolor':'midnightblue',
'pad':5},
            color = 'w',fontsize=18)
#


---


# I set the legend: abbreviation -> airline name
comp_handler = []
for i in range(len(abbr_companies)):
    comp_handler.append(mpatches.Patch(color=colors[i],
label = global_stats.index[i] + ': ' +
abbr_companies[global_stats.index[i]]))
ax1.legend(handles=comp_handler, bbox_to_anchor=(0.2, 0.9),
          fontsize = 13, bbox_transform=plt.gcf().transFigure)
#-----

```

```

# Pie chart nº2: mean delay at departure
#-----
sizes = global_stats['mean'].values
sizes = [max(s,0) for s in sizes]
explode = [0.0 if sizes[i] < 20000 else 0.01 for i in
range(len(abbr_companies))]
patches, texts, autotexts = ax2.pie(sizes, explode = explode, labels = labels,
                                colors = colors, shadow=False, startangle=0,
                                autopct = lambda p : '{:.0f}'.format(p * sum(sizes) / 100))
for i in range(len(abbr_companies)):
    texts[i].set_fontsize(14)
ax2.axis('equal')
ax2.set_title('Mean delay at origin', bbox={'facecolor':'midnightblue', 'pad':5},
            color='w', fontsize=18)
#-----
# stripplot with all the values reported for the delays
# _____

# I redefine the colors for correspondance with the pie charts
colors = ['firebrick', 'gold', 'lightcoral', 'aquamarine', 'c', 'yellowgreen', 'grey',
        'seagreen', 'tomato', 'violet', 'wheat', 'chartreuse', 'lightskyblue',
        'royalblue']
# _____

ax3 = sns.stripplot(y="AIRLINE", x="DEPARTURE_DELAY", size = 4, palette
= colors,
                    data=df2, linewidth = 0.5, jitter=True)
plt.setp(ax3.get_xticklabels(), fontsize=14)
plt.setp(ax3.get_yticklabels(), fontsize=14)
ax3.set_xticklabels(['{:.20f}h{:2.0f}m'.format(*[int(y) for y in divmod(x,60)])
                    for x in ax3.get_xticks()])
plt.xlabel('Departure delay', fontsize=18, bbox={'facecolor':'midnightblue',
'pad':5},
            color='w', labelpad=20)
ax3.yaxis.label.set_visible(False)
# _____
plt.tight_layout(w_pad=3)

# _____
# Function that define how delays are grouped
delay_type = lambda x:((0,1)[x > 5],2)[x > 45]
df['DELAY_LEVEL'] = df['DEPARTURE_DELAY'].apply(delay_type)
# _____
fig = plt.figure(1, figsize=(10,7))

```

```

ax = sns.countplot(y="AIRLINE", hue='DELAY_LEVEL', data=df)
# _____

# We replace the abbreviations by the full names of the companies and set
the labels
labels = [abbr_companies[item.get_text()] for item in ax.get_yticklabels()]
ax.set_yticklabels(labels)
plt.setp(ax.get_xticklabels(), fontsize=12, weight = 'normal', rotation = 0);
plt.setp(ax.get_yticklabels(), fontsize=12, weight = 'bold', rotation = 0);
ax.yaxis.label.set_visible(False)
plt.xlabel('Flight count', fontsize=16, weight = 'bold', labelpad=10)
# _____
# Set the legend
L = plt.legend()
L.get_texts()[0].set_text('on time (t < 5 min)')
L.get_texts()[1].set_text('small delay (5 < t < 45 min)')
L.get_texts()[2].set_text('large delay (t > 45 min)')
plt.show()

# _____
# Model function used to fit the histograms
def func(x, a, b):
    return a * np.exp(-x/b)
#-----
points = [] ; label_company = []
fig = plt.figure(1, figsize=(11,11))
i = 0
for carrier_name in [abbr_companies[x] for x in global_stats.index]:
    i += 1
    ax = fig.add_subplot(5,3,i)
    # _____
    # Fit of the distribution
    n, bins, patches = plt.hist(x =
df2[df2['AIRLINE']==carrier_name]['DEPARTURE_DELAY'],
                                range = (15,180), normed=True, bins= 60)
    bin_centers = bins[:-1] + 0.5 * (bins[1:] - bins[:-1])
    popt, pcov = curve_fit(func, bin_centers, n, p0 = [1, 2])
    # _____
    # bookkeeping of the results
    points.append(popt)
    label_company.append(carrier_name)
    # _____
    # draw the fit curve
    plt.plot(bin_centers, func(bin_centers, *popt), 'r-', linewidth=3)

```

```

# _____
# define tick labels for each subplot
if i < 10:
    ax.set_xticklabels([" for x in ax.get_xticks()])
else:
    ax.set_xticklabels(['{:2.0f}h{:2.0f}m'.format(*[int(y) for y in divmod(x,60)])
                        for x in ax.get_xticks()])
# _____
# subplot title
plt.title(carrier_name, fontsize = 14, fontweight = 'bold', color = 'darkblue')
# _____
# axes labels
if i == 4:
    ax.text(-0.3,0.9,'Normalized count of flights', fontsize=16, rotation=90,
            color='k', horizontalalignment='center', transform = ax.transAxes)
if i == 14:
    ax.text( 0.5, -0.5 , 'Delay at origin', fontsize=16, rotation=0,
            color='k', horizontalalignment='center', transform = ax.transAxes)
# _____
# Legend: values of the a and b coefficients
ax.text(0.68, 0.7, 'a = {} \nb = {}'.format(round(popt[0],2), round(popt[1],1)),
        style='italic', transform=ax.transAxes, fontsize = 12, family='fantasy',
        bbox={'facecolor':'tomato', 'alpha':0.8, 'pad':5})

plt.tight_layout()

```

B. SCREEN SHOTS

