International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

Airport Data Interpretation using Python Programming

Nasvi Kareem

Data Researcher, Department of Technology, Learners Point Training Academy

nasvikareem.mec[at]gmail.com

Abstract: ***Data is information about a subject dealing in an organization which can be in structured or unstructured format. There are many methods to extract data from different sources. Data Analysis consists of different steps- Data Extraction/Data filtering/ Data plotting/ Data prediction. This project gives an understanding on how to execute data analysis by using python programming. Python is a popular language used widely. Python is a flexible language which can be executed in different platforms. Python has libraries which support data analysis. The project explains the step by execution of python language from loading the multiple files and giving a result of different plots. Tkinter is a python user interface used to display the results in an accountable-application manner. The codes are executed in Jupyter Notebook using the relevant python libraries. At the conclusion the plots were displayed from the three categories of airports where the location is filtered to display only some regions of flight landing. It works on a graphical user interface which is generated as an application using buttons which display the output.***

Keywords: Python, Jupyter Notebook, Data Filtering, Data Plotting, Data Visualization, Tkinter, Numpy, Pandas, Matplotlib, Seaborn

1. Introduction

Data is a piece of information collected by raw means which must be converted into meaningful information with relevant means. To carry out this process of conversion the concept of data analysis has been growing extensively. In the earlier years, huge amounts of data were not produced. Very small-scale data were produced and could be processed by traditional methods. But as technology advanced, through the years, the accumulation of data has increased tremendously and will lead to the concept of data science.

Data analysis is a process of extracting data from different sources. It involves many methodologies like data cleaning, data filtering, data plotting. The main agenda behind the analytics is to extract relevant information from multiple datasets to customize the requirements. The process of data analysis contains a step wise execution and can be performed through any analytical tools like-Microsoft Excel/ Power Pi or Tableau. Data analysis can be executed by python programming language as it has libraries to support the analysis.

Data cleaning is the process of removing unwanted data from sources. The unwanted data can be in the form of null values or. This is a very important step as the data cleaning makes the data into a structured format in the form of proper rows and columns with relevant details in it. Data filtering follows the next step after cleaning. Filtering helps to select the data required to reach the end goal by eliminating the columns not usable for the goal. Filtering can be performed by either hiding the non-usable details or just eliminating it completely from the dataset. Data plotting involves the graphical representation of the tabular content into a presentable manner to make analysis and conclusions about the dataset. Data visualization provides different representations like line chart, bar-chart, pie-plots, boxplots, joint-plots and many more using the different combination of columns.

Python is a programming language used in data science. Python has libraries which support the analytics and brings a valuable result to the requirement. It is an open-source platform which can load any format of file without any challenges. Various file formats like Excel, CSV, XML.

HTML and JSON file formats are compatible with python loading.

Python involves libraries like Numpy, Pandas, Matplotlib, Seaborn, Tkinter python library is used to bring out a graphical interface to the project and it generates various outputs by clicking on the buttons created in the program. This interphase is free open python extension and is widely used to generate applications for different implementations.
Data Analysis with Python is an easy technique to process and transform large amounts of data into proper subsets and generate meaningful outputs by simply importing libraries. This method is adopted as it helps to reduce the chance of errors and it also provides an easy path for the analysis.

There is an easy flow for the project, which helps to understand the agenda of the project in the best way. Data analysis can be done in different methods like using data analytics Tools like Power Bi, Tableau or MS excel. But this project shows the easiest method to collect and transform the data.

2. Methodology

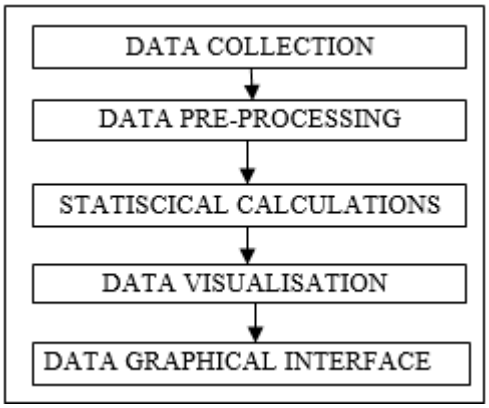

Figure1: Flow chart of the project

2.1 Data collection

Data collection is the process of gathering out data's from different sources and converting them into formats for loading them.

In this project, three sets of CSV files are taken for loading- Airports, Airport-Frequencies and Runway CSV files. Reading the csv file can be performed by python syntax. The files are loaded into three Data frames and are ready for the data pre-processing stage. The files were loaded to give a good analysis based on the different airports in different countries which included small airports, heliport, medium airport, large airports, and closed airports. The airport frequency file gave the relevant information about different frequencies of loading in different airports and runway file gave the details about the runway which includes latitude, longitude and many more columns included in it.

2.2 Data Pre-Processing Stage

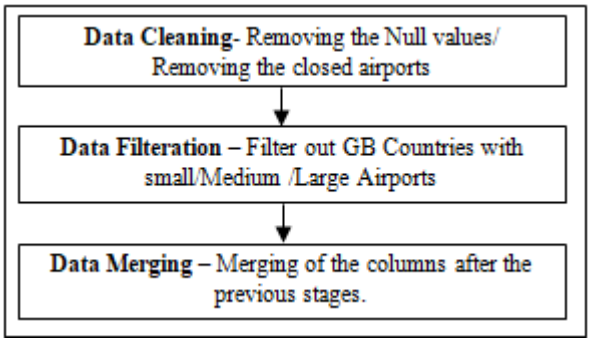

Figure 2: Flow chart of Data pre-processing stage

Data Preprocessing stage includes data cleaning, removing the unwanted cells or columns, clearing out the duplicated data in the files. Data cleaning has been performed by checking if any null values are present in the datasets. The airport file was checked for null values and there was a significant count of null values in a few columns. The null values were replaced by frequently occurring values from the columns. This was performed by using mode instruction and the process of replacement was complete. All the closed airports from the three files were deleted to create a good analysis. All the statistical counts from different files were analyzed and stored for observation.

In the second stage of data processing, the process of filtration was performed by filtering out airports which fall into the category of Great Britain (GB). Data filteration is the process of filtering out unwanted rows and columns from the requirement to sort out the information only bringing out the best results. From GB Airports, the second filteration was performed to separate out the small, medium and large airports from the data sets separately and they were saved into different data frames.

In the third stage of data processing the three files were merged to analyze and get the associated rows and columns in one single table. Finally the three merged files included the details of the columns – ID, Ident, Type (Small/ Medium, Large) , Country(GB), closed Counts. This was separately created as small airport under GB, Medium Airport under GB, Large Airport Under GB. The files were checked again to find if all the closed airports are removed from the new data frames.

2.3 Statistical Calculations

The three files generated were made to perform some statistical interpretation to display some values. The mean, median and mode were calculated for three airports based on their frequencies represented in MHz Those airports from each sets, small, medium and large, the frequencies fall above 100 MHZ were filtered and their mean value was also calculated separately

2.4 Data Visualization

Data Visualization is the graphical representation of data and information. By using various tools, we can visualize the data into meaningful graphs and plots to fill in the dashboard. Elements like charts, plots, and maps are used to present the datasets.

Based on the count of frequencies and the frequency ranges three scatter plots were created separately for small, large, and medium airports.

2.5 Data Graphical Interface

Data interface is a user graphical representation of the application to view the results in a better way. The application can include buttons, widgets, text boxes, labels, charts and tables to represent the interpretations from the datasets. Python has a lot of GUI Frameworks, but Tkinter is the only framework built from the standard library. Tkinter Python Library is used to represent the three data sets into the collaboration with the interface.

Tkinter is an open-source Python Library which is an easy and simple application. It provides a powerful object-oriented interface to the Tk GUI Toolkit. Importing Tkinter is the same way as other libraries are imported for performing the task. In this project, the results were displayed using Tkinter library by creating buttons and displaying the charts and tables obtained from the data preprocessing and visualization steps.

3. Python Implementation and Execution

3.1 Platform Selection

Jupyter Notebook is an easy and flexible platform used to perform projects under Data science. The interface of Jupyter is not very complicated to load and execute any types of file formats. The platform even supports the loading of data from SQL Servers which becomes an important step in Data Loading.

In this project, three CSV Format files were loaded onto the platform. Each Python file works under the version PYTHON-3 and python codes were written into different

cells in the platform to view the results separately. Each file was loaded into data frames and then further processed for data analysis.

3.2 Importing Libraries

Python language is an open-source programming concept which has many extended libraries which helps to perform the data processing steps. Below figure shows the loading of the python libraries into the platform.

The figure explaining importing of libraries like JSON, CSV, Pandas, Numpy, Matplotlib, Seaborn and Tkinter.

```
#library import for Json
import json, csv
from tkinter import filedialog
#library import for any path
from tkinter import *
#library import for message box
from tkinter import messagebox
from tkinter.filedialog import asksaveasfile
#library import for gui
import tkinter as tk
import matplotlib
#library import fior pandas table display
from pandastable import Table, TableModel
#library import for plot
import matplotlib.pyplot as plt
matplotlib.use('TkAgg')
#library import for images
from matplotlib.figure import Figure
#library import for canvas
from matplotlib.backends.backend_tkagg import (FigureCanvasTkAgg,NavigationToolbar2Tk)
import seaborn as sns
import pandas as pd
#library import for numerical data
import numpy as np
```

Figure 3: Importing the Python Libraries

3.3 Loading of files

The datasets were loaded into the platform and represented in form of Data Frames with the Pandas library. These Data Frames were later on converted into JSON Formats as a end result.

```
df = pd.read_csv ('airports.csv')
de=pd.read_csv ('airport-frequencies.csv')
dg=pd.read_csv ('runways.csv')
#Creating the datatframe
df3= pd.DataFrame(df)
de3=pd.DataFrame(de)
dg3=pd.DataFrame(dg)
```

Figure 4: Loading the CSV Files

The figure explains the loading of the datasets using pd.read_csv syntax and represented into df3,de3 and dg3 respectively.

3.4 Understanding the datasets

The loaded datasets were observed and anlysed using the python synatx. This step is cruicial step to analyse what and where the corrections has to be implemented into the dataet inorder to get best results.

```
#shape of the dataframe
df3.shape
```

```
(68947, 18)
```

Figure 5: Displaying the count of rows and columns

Python gives many syntax to support this like shape(), describe(), info() ,head() and tail(). Below figure explains few of the analysis results.

```
df3.describe()
```

	id	latitude_deg	longitude_deg	elevation_ft
count	68947.000000	68947.000000	68947.000000	55871.000000
mean	135599.402469	25.976450	-31.077357	1287.082422
std	149239.463407	26.293894	84.607771	1648.526320
min	2.000000	-90.000000	-179.876999	-1266.000000
25%	17373.500000	12.509784	-94.178001	207.000000
50%	37104.000000	35.397301	-71.007500	729.000000
75%	324260.500000	42.917049	19.420111	1585.000000
max	349799.000000	82.750000	179.975700	22000.000000

```
df3.columns.values
```

```
array(['id', 'ident', 'type', 'name', 'latitude_deg', 'longitude_deg',
       'elevation_ft', 'continent', 'iso_country', 'iso_region',
       'municipality', 'scheduled_service', 'gps_code', 'iata_code',
       'local_code', 'home_link', 'wikipedia_link', 'keywords'],
      dtype=object)
```

Figure 6: Displaying the statistical count and column names

3.5 Data Cleaning

Data cleaning was performed as there were many empty cells in the dataframes. It not only includes removal of empty cells but also deleting the unwanted rows or columns which were not to be significantly used in the project.

```
result.isnull().sum()
```

```
id                    0
ident                 0
type                  0
name                  0
latitude_deg          0
longitude_deg         0
elevation_ft          0
continent             0
iso_country           0
iso_region            0
municipality          0
scheduled_service     0
gps_code              0
iata_code             0
local_code            0
home_link             0
wikipedia_link        0
keywords              0
dtype: int64
```

Figure 7: Removed all null values

It was significantly observed that most of the columns had null values in all three datasets , hence it was replaced with significantly occuring value from each column which helped to eleiminate all the null values and give the results as below. Figure explains the nullification of all empty cells from the datasets.

3.6 Data Filteration

Data filteration includes filtering out unwanted columns. Closed Airports were deleted from all the dataset to avoid reduction in accuracy in the project.

```
dg3.drop(dg3[dg3['closed'] == 1].index, inplace = True)
dg3
```

	id	airport_ref	airport_ident	length_ft	width_ft	surface	lighted	closed	le_ident
0	269408	6523	00A	80.0	80.0	ASPH-G	1	0	H1
1	255155	6524	00AK	2500.0	70.0	GRVL	0	0	N
2	254165	6525	00AL	2300.0	200.0	TURF	0	0	01
3	270932	6526	00AR	40.0	40.0	GRASS	0	0	H1
4	322128	322127	00AS	1450.0	60.0	Turf	0	0	1
...	...	...	...	...	...	...	...	...	...
42872	235188	27242	ZYTL	10827.0	148.0	CON	1	0	10
42873	235186	27243	ZYTX	10499.0	148.0	Asphalt	1	0	06
42874	235169	27244	ZYYJ	8530.0	148.0	CON	1	0	09
42875	346789	346788	ZZ-0003	1800.0	15.0	Turf	0	0	15
42876	313663	313629	ZZZZ	1713.0	82.0	concrete	0	0	18

42185 rows × 21 columns

Figure 8: Removing the closed Airports

The airports belonging to Country GB-Great Britain were only chosen as per the requirement.

```
filter2=df3[(df3.type == 'medium_airport') & (df3.iso_country == "GB")  ]
filter2
```

	id	ident	type	name	latitude_deg	longitude_deg	elevation_ft
20868	2386	EGAB	medium_airport	Enniskillen/St Angelo Airport	54.398899	-7.651670	155.0
20869	2387	EGAC	medium_airport	George Best Belfast City Airport	54.618099	-5.872500	15.0
20871	2388	EGAE	medium_airport	City of Derry Airport	55.042801	-7.161110	22.0
20879	2390	EGBE	medium_airport	Coventry Airport	52.369701	-1.479720	267.0
20882	2392	EGBJ	medium_airport	Gloucestershire Airport	51.894199	-2.167220	101.0

Figure 9: Filtering out GB Airports

3.7 Data Visualization for the three airports

Data plotting was executed using the libraries- Matplotlib and Seaborn. Scatter plots were generated based on the frequency details from the datasets. Three plots were generated based on three types of airports- small, medium, and large Airports all falling under the GB Category.

The figure explains the colors styles and legends display for the three airports.

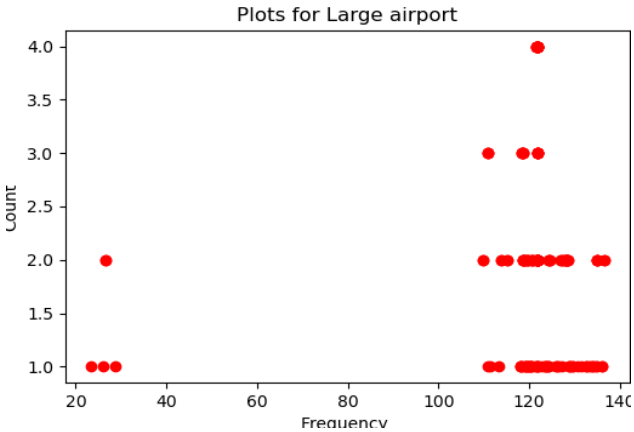

Figure 11: Scatter plots- Large Airports(GB)

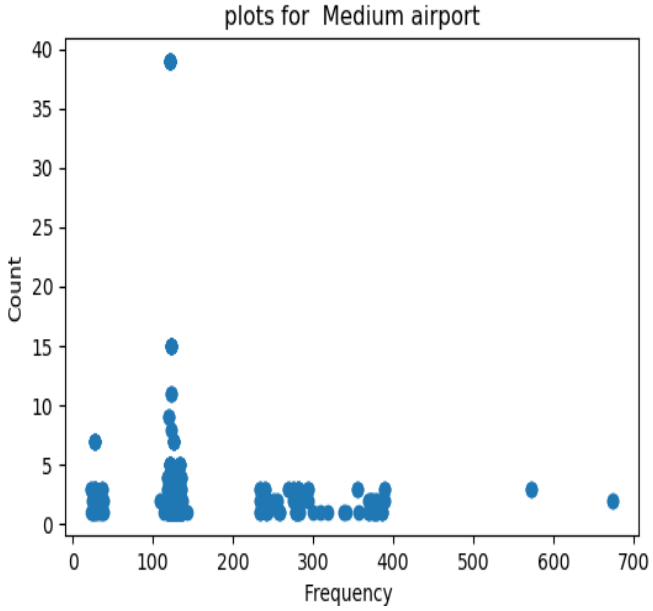

Figure 12: Scatter plots-Medium Airports(GB)

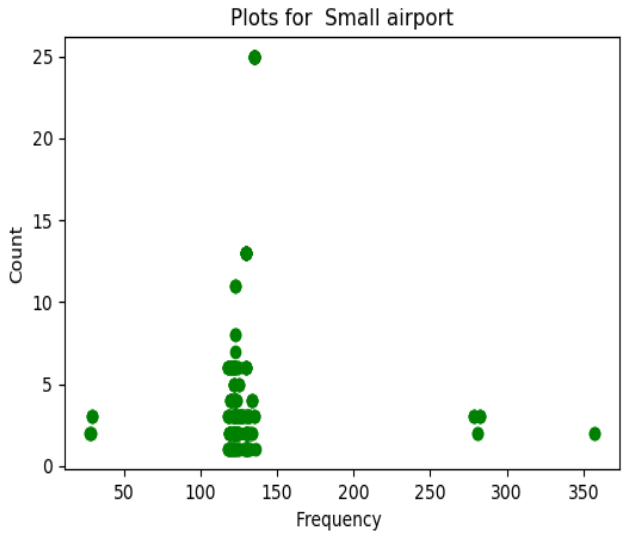

Figure 13: Scatter plots- Small Airports(GB)

Scatter plots were chosen as the point signifies the count and frequencies matching to each data cell which helps to take a clear picture of the number of airports with their sizes falling under the desired category. Adding on to the plots, scatter plots were represented with thick dots for better visualization in the tkinter.

3.8 Tkinter User Interface

Tkinter is the graphical user interface used in the project. This was chosen being the easiest interface for representation and the buttons and labels made the project look visually better when compared to the other interface. This is the only interface where the standard library of Python was used for implementation.

The figure explains the various buttons inserted in the project to choose and find the results as per the requirement. This works like an application where the insertion of three CSV Files were given to the loader using the import button. The clean CSV button cleans the null values and closed airports and filtered the datasets based on GB Countries falling under the three categories of airports- Small, Medium and Large. Mean/ Median and mode button gave the frequency values of large airports.

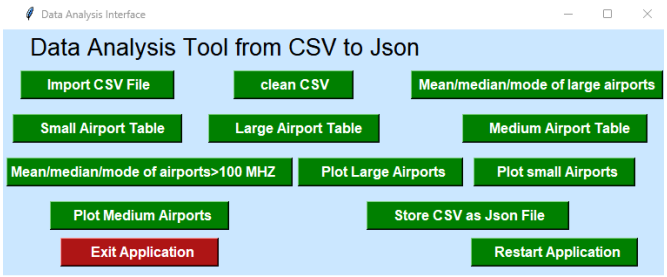

Figure 14: Graphical User Interface

Three buttons were labelled to display the final small, medium, and large dataset along with separate buttons for the plots. Finally, the dataset was converted into JSON File formats and saved into the path. There were button-Restart and exit applications if one chose to restart the process from the start and even exit the application if they do not want to display any more results. The project was carried out step by step to get the three divisions of airports in the functional manner.

4. Conclusion

Data science has become one of the important aspects of data analytics. Data analysts can execute the analysis in different manners. This project gives a detailed description about the possibility to analyze the datasets and represent it in a simple processing stage. Python is an open-source platform which provides the flexibility to choose and import any library without any limitation. Python programming helped to carry out the analysis in a simple way and the project could even be loaded into any data science tools for further changes. The project works on an application based where the interesting section was insertion of interface which helped to understand the three datasets in a refined manner. The project flow included simple steps of data extraction, filteration and visualization and conversion of file format. The file can be chosen to be converted into any format to extend the analysis. The scatter plot was chosen which added better dashboard result to the project. The results were displayed using a suitable interface where the clicking action generated results from different sections.

References

[1] A. Nagpal and G. Gabrani, "Python for Data Analytics, Scientific and Technical Applications," *2019 Amity International Conference on Artificial Intelligence (AICAI)*, 2019, pp. 140-145, doi: 10.1109/AICAI.2019.8701341.
[2] Kiranbala Nongthombam, “Data Analysis using Python”, International Journal of Engineering Research & Technology, IJERTV10IS070241, Vol. 10 Issue 07, July-2021.
[3] V. I. Aladesanmi, O. S. Fatoba, T. C. Jen and E. T. Akinlabi, "Python Data Analysis and Regression Plots of Wear and Hardness Characteristics of Laser Cladded Ti and TiB2 Nanocomposites on Steel Rail," *2021 IEEE 12th International Conference on Mechanical and Intelligent Manufacturing Technologies (ICMIMT)*, 2021, pp. 40-44, doi: 10.1109/ICMIMT52186.2021.9476211.
[4] X. Liu and H. Xu, "School-Enterprise Cooperation on Python Data Analysis Teaching," *2019 14th International Conference on Computer Science & Education (ICCSE)*, 2019, pp. 278-281, doi: 10.1109/ICCSE.2019.8845524.
[5] T. Surya Gunawan, N. Aleah Jehan Abdullah, M. Kartiwi and E. Ihsanto, "Social Network Analysis using Python Data Mining," *2020 8th International Conference on Cyber and IT Service Management (CITSM)*, 2020, pp. 1-6, doi: 10.1109/CITSM50537.2020.9268866.
[6] Wes McKinney, “Python for Data Analysis, 2nd Edition”, Released October 2017, O'Reilly Media, Inc.