LUNG CANCER PREDICTION

Project Deliverable Part 4

Introduction

In this project we will analyze the lung cancer data. The dataset is gathered from the online lung cancer prediction system website. Hypotheses of the include:

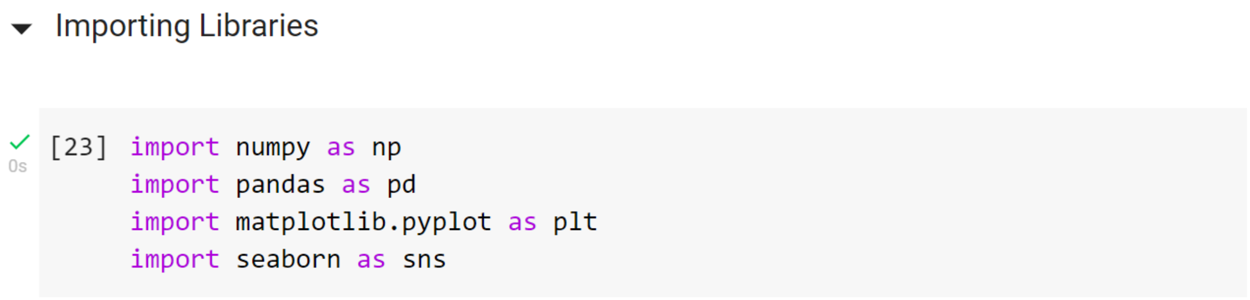
1. people who smoke have the greatest risk of lung cancer, though cancer can also occur in people who have never smoked
2. The risk of lung cancer increases with the length of time and number of cigarettes someone's smoked.
3. If someone quits smoking, even after smoking for many years, he/she can significantly
 reduce the chances of developing lung cancer

As part of this project, we will also analyze which machine learning model gives best accuracy and efficiency.

Prepare and Process Data

The lung cancer dataset contains 16 attributes and 309 entries. First step of the project is to download the data and analyze it. I used a spreadsheet for the first step analysis to understand the attribute names and the data type of each attribute and also python jupyter notebook for processing the dataset.

Importing all the necessary libraries before processing and cleaning the dataset.

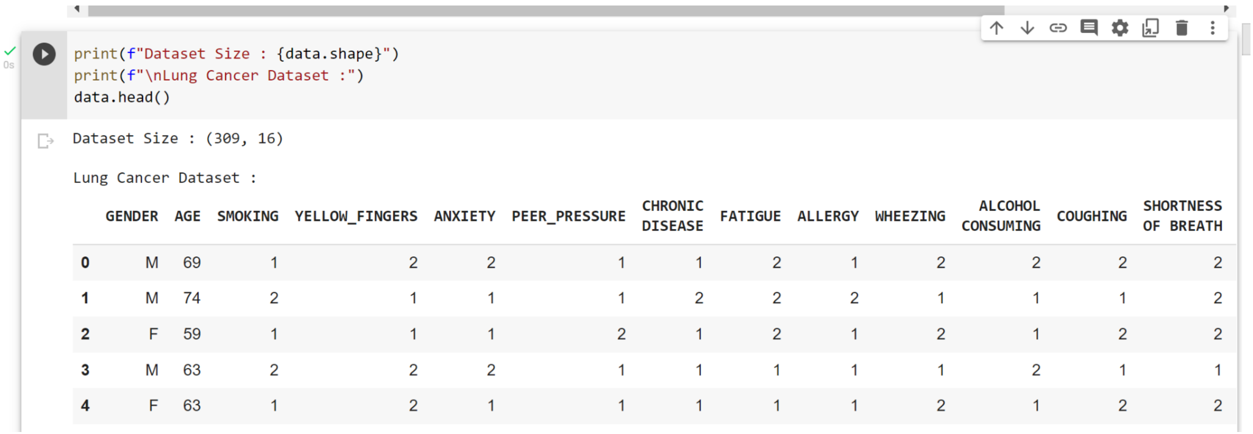


▾ Importing Libraries

```
[23] import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
     import seaborn as sns
```

Tool: python, Jupyter Notebook

Having a glimpse over the dataset imported to jupyter notebook.



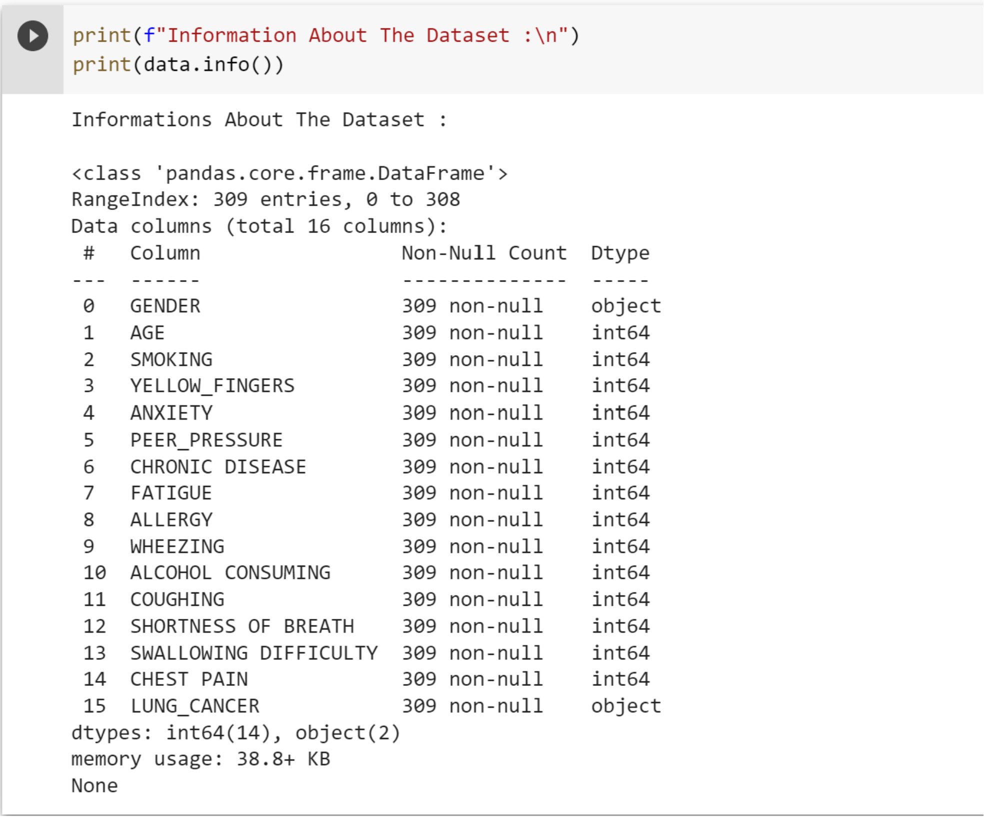
```
print(f"Dataset Size : {data.shape}")
print(f"\nLung Cancer Dataset :")
data.head()
```

```
Dataset Size : (309, 16)

Lung Cancer Dataset :
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH
0	M	69	1	2	2	1	1	2	1	2	2	2	2
1	M	74	2	1	1	1	2	2	2	1	1	1	2
2	F	59	1	1	1	2	1	2	1	2	1	2	2
3	M	63	2	2	2	1	1	1	1	1	2	1	1
4	F	63	1	2	1	1	1	1	1	2	1	2	2

Tool: python, Jupyter Notebook

Information about the dataset to check attribute names and data types.



```
print(f"Information About The Dataset :\n")
print(data.info())
```

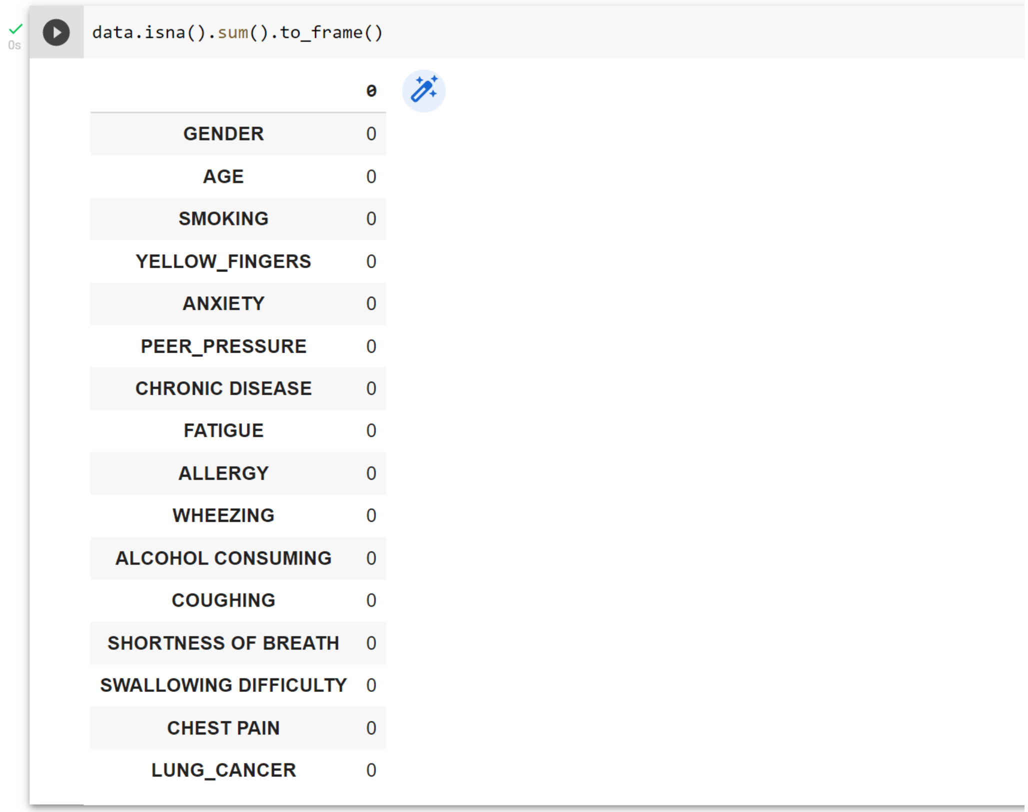
```
Informations About The Dataset :

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 309 entries, 0 to 308
Data columns (total 16 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   GENDER                 309 non-null    object
 1   AGE                    309 non-null    int64
 2   SMOKING                309 non-null    int64
 3   YELLOW_FINGERS         309 non-null    int64
 4   ANXIETY                309 non-null    int64
 5   PEER_PRESSURE          309 non-null    int64
 6   CHRONIC DISEASE        309 non-null    int64
 7   FATIGUE                309 non-null    int64
 8   ALLERGY                309 non-null    int64
 9   WHEEZING               309 non-null    int64
 10  ALCOHOL CONSUMING      309 non-null    int64
 11  COUGHING               309 non-null    int64
 12  SHORTNESS OF BREATH    309 non-null    int64
 13  SWALLOWING DIFFICULTY  309 non-null    int64
 14  CHEST PAIN             309 non-null    int64
 15  LUNG_CANCER            309 non-null    object
dtypes: int64(14), object(2)
memory usage: 38.8+ KB
None
```

Tool: python, Jupyter Notebook

Cleaning data

The first step in the process of cleaning data is to check if the data has any null values. Having null values in the dataset creates ambiguity and it will lead to improper conclusions if we train the algorithms with data which contains null values. Hence, it is really important to check and handle null values.

Here, the lung cancer dataset does not contain null values.

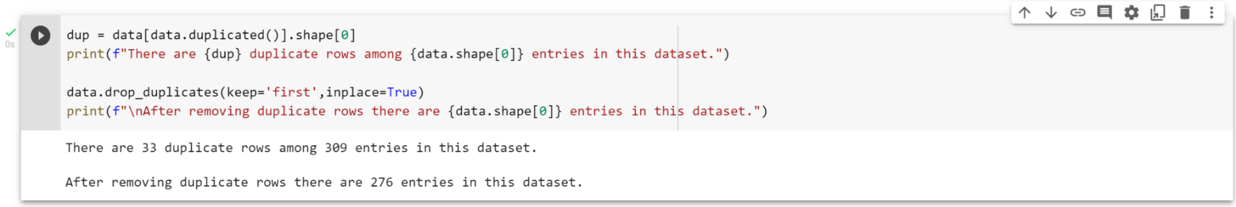


```
data.isna().sum().to_frame()
```

	0
GENDER	0
AGE	0
SMOKING	0
YELLOW_FINGERS	0
ANXIETY	0
PEER_PRESSURE	0
CHRONIC DISEASE	0
FATIGUE	0
ALLERGY	0
WHEEZING	0
ALCOHOL CONSUMING	0
COUGHING	0
SHORTNESS OF BREATH	0
SWALLOWING DIFFICULTY	0
CHEST PAIN	0
LUNG_CANCER	0

Tool: python, Jupyter Notebook

The second step in cleaning is to check if the data set has duplicate values. Duplicate data sets run the risk of contaminating the training data with the test data, or the other way around.



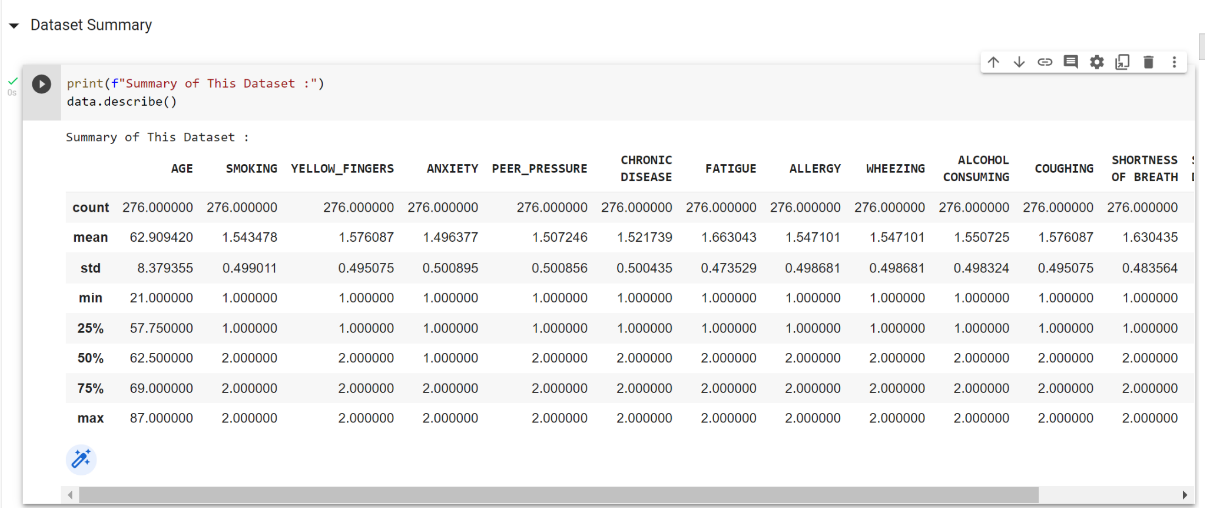
```
dup = data[data.duplicated()].shape[0]
print(f"There are {dup} duplicate rows among {data.shape[0]} entries in this dataset.")

data.drop_duplicates(keep='first',inplace=True)
print(f"\nAfter removing duplicate rows there are {data.shape[0]} entries in this dataset.")
```

```
There are 33 duplicate rows among 309 entries in this dataset.

After removing duplicate rows there are 276 entries in this dataset.
```

Tool: python, Jupyter Notebook

Exploring Data

As part of exploring the dataset, I dug deeper into the data and analyzed it. As a first step, I found the summary of data using the describe() function in python.



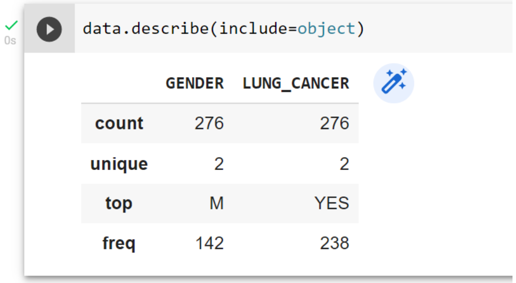
Dataset Summary

```
print(f"Summary of This Dataset :")
data.describe()
```

```
Summary of This Dataset :
```

	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH
count	276.000000	276.000000	276.000000	276.000000	276.000000	276.000000	276.000000	276.000000	276.000000	276.000000	276.000000	276.000000
mean	62.909420	1.543478	1.576087	1.496377	1.507246	1.521739	1.663043	1.547101	1.547101	1.550725	1.576087	1.630435
std	8.379355	0.499011	0.495075	0.500895	0.500856	0.500435	0.473529	0.498681	0.498681	0.498324	0.495075	0.483564
min	21.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
25%	57.750000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
50%	62.500000	2.000000	2.000000	1.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000
75%	69.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000
max	87.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000

Tool: python, Jupyter Notebook



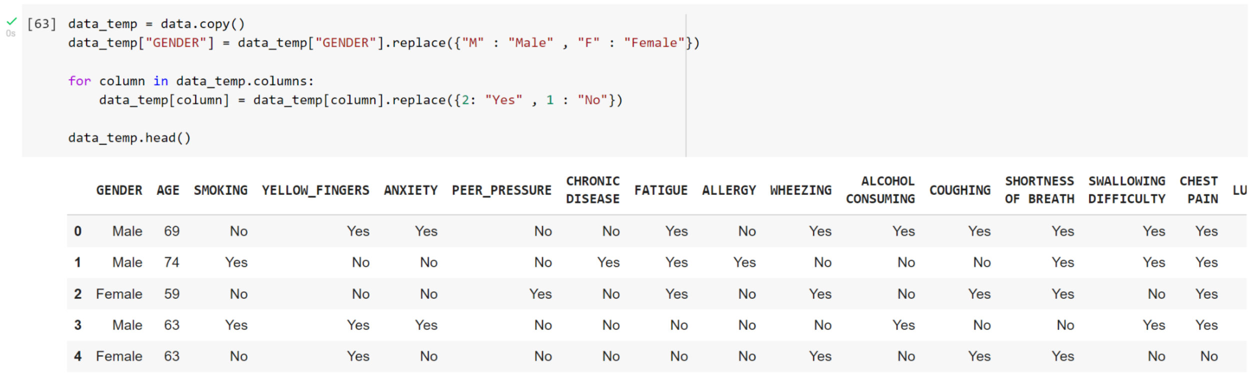
```
data.describe(include=object)
```

	GENDER	LUNG_CANCER
count	276	276
unique	2	2
top	M	YES
freq	142	238

The above description shows that there are 276 rows with 2 unique values for both Gender(male, female) and Lung_cancer(yes,no) and there are 142 rows with gender as male and 238 rows with positive lung cancer.

VISUALIZATION

For Better visualization, I replaced all the numerical values into categorical values and replaced ‘M’ and ‘F’ in Gender attribute to ‘Male’ and ‘Female’. Similarly, I replaced 1 and 2 with yes and no.

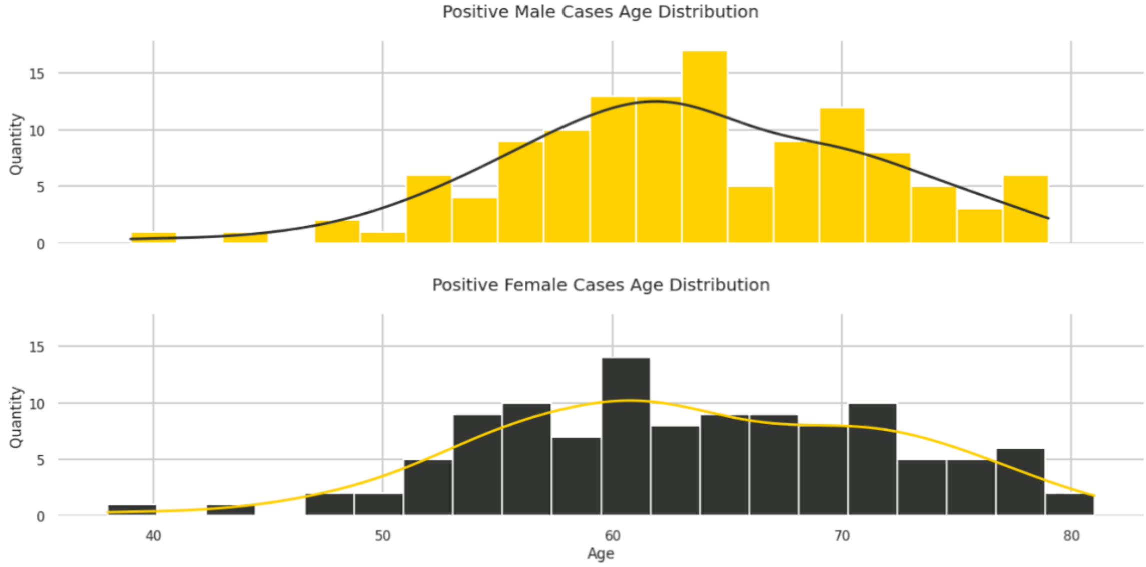


```python
data_temp = data.copy()
data_temp["GENDER"] = data_temp["GENDER"].replace({"M" : "Male" , "F" : "Female"})

for column in data_temp.columns:
    data_temp[column] = data_temp[column].replace({2: "Yes" , 1 : "No"})

data_temp.head()
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH	SWALLOWING DIFFICULTY	CHEST PAIN	LU
0	Male	69	No	Yes	Yes	No	No	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	
1	Male	74	Yes	No	No	No	Yes	Yes	Yes	No	No	No	Yes	Yes	Yes	
2	Female	59	No	No	No	Yes	No	Yes	No	Yes	No	Yes	Yes	No	Yes	
3	Male	63	Yes	Yes	Yes	No	No	No	No	No	Yes	No	No	Yes	Yes	
4	Female	63	No	Yes	No	No	No	No	No	Yes	No	Yes	Yes	No	No	

Tool: python, Jupyter Notebook

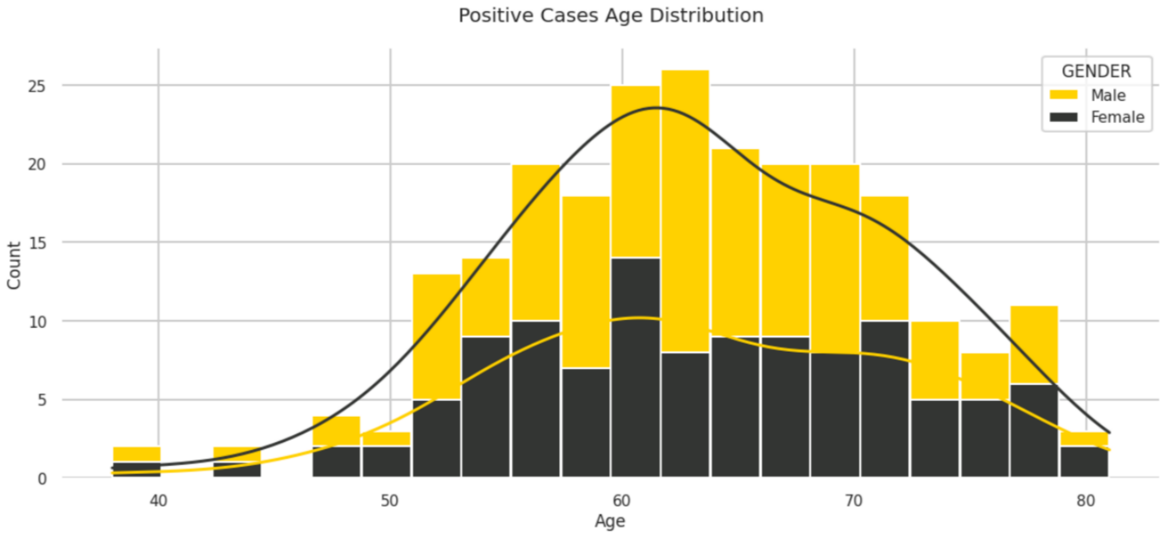
As part of visualization, I dug deeper into the data as an initial step and analyzed it. In the lung cancer dataset there will be both positive and negative cases. I have analyzed the positive cases and I found the correlation between positive cases and age for both male and female lung cancer prediction cases.



Tool: python, Jupyter Notebook

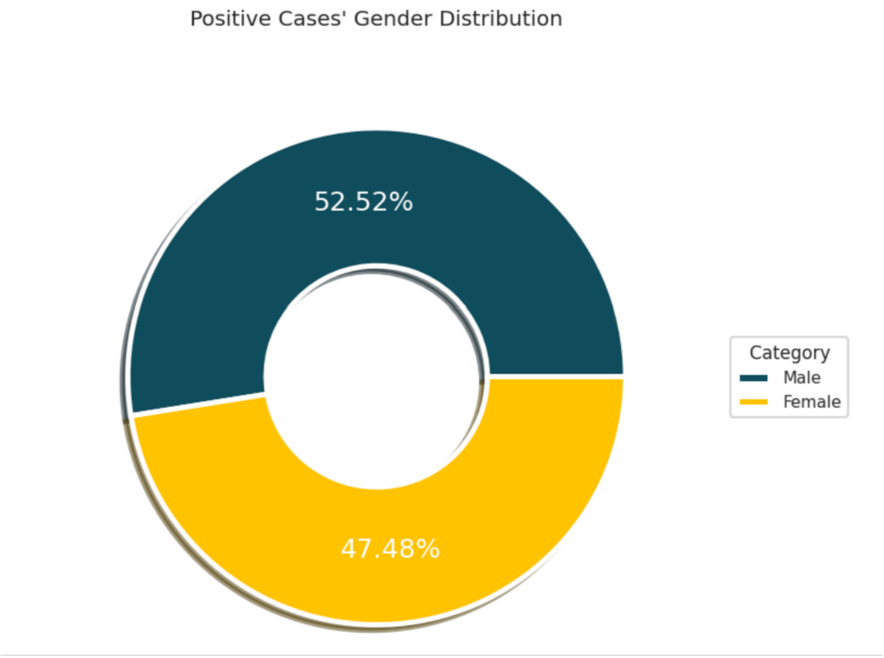
For Better visualization, I tried to club both the graphs into a single graph by stacking them together.

Tool: python, Jupyter Notebook

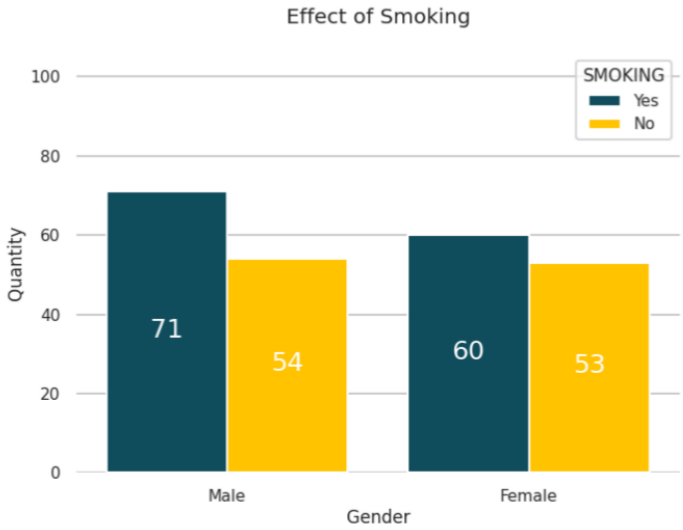


The next important visualization is based on gender,I tried to analyze if there is any relation between gender and the number of positive cases for lung cancer. It turned out that males are more prone to lung cancer than females with 52.52% and females being at 47.48.

Tool: python, Jupyter Notebook

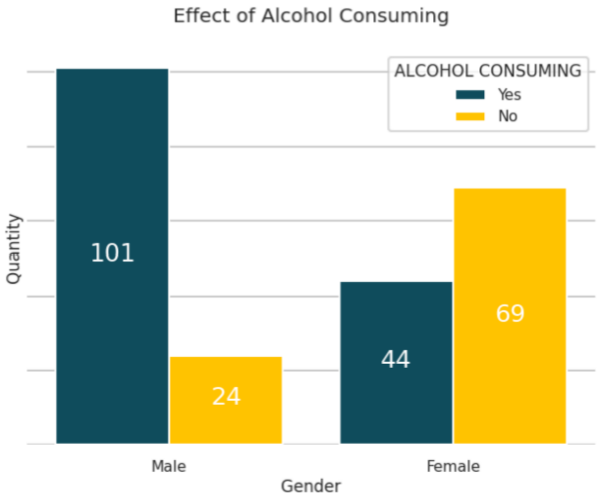


According to my hypotheses, I wanted to explore the relation between smoking, alcohol and the number of positive cases.



Tool: python, Jupyter Notebook

The results that I achieved aligned with our hypothesis. According to the graph, people who have done smoking have higher risk of getting lung cancer and it also shows that people who do not smoke will also be affected by lung cancer but which is less when compared to the people who smoke.



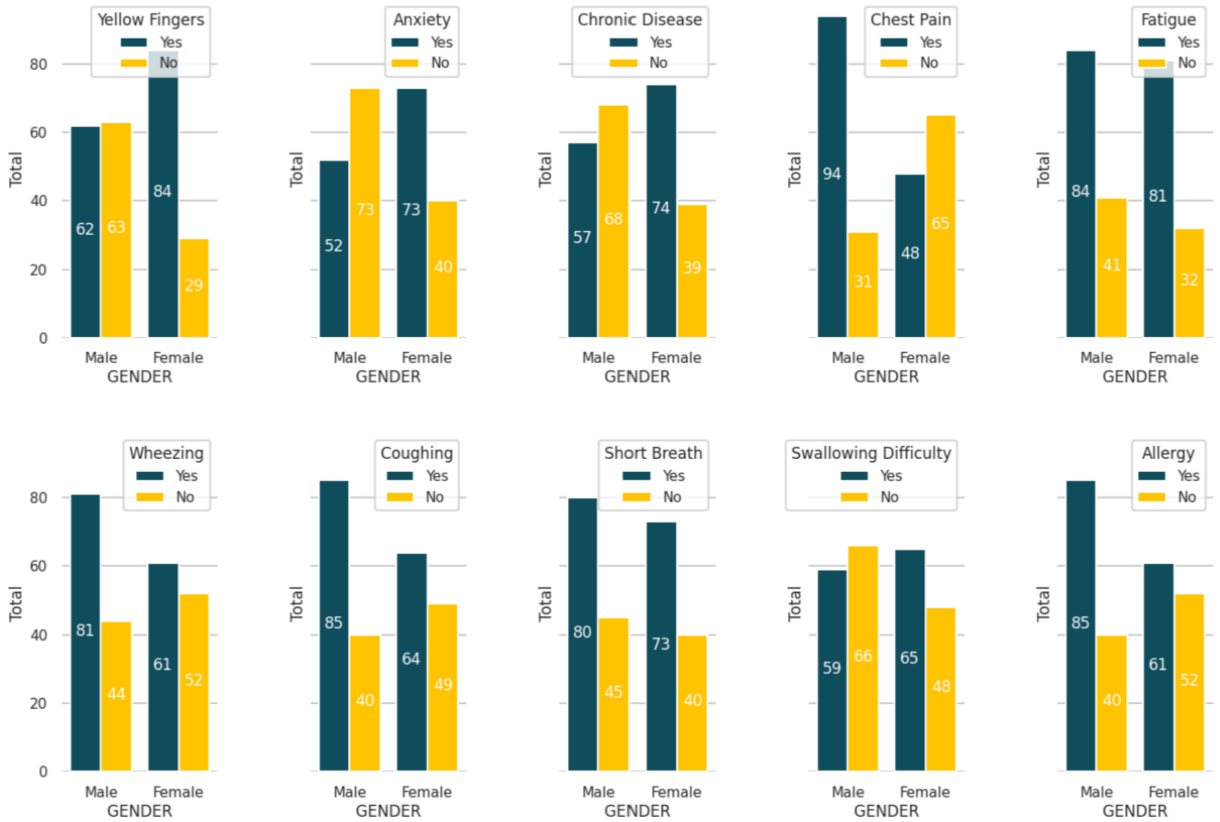
Tool: python, Jupyter Notebook

I have continued the similar type of analysis between alcohol and the number of positive cases for lung cancer. To my surprise, I have got interesting results for this analysis.
In the case of males, most of the positive cases are from persons who consume alcohol regularly and in case of females, a high amount of positive cases are from the persons who do not consume alcohol.

As part of visualization itself, I have worked a bit further to understand more and dig deeper into the data, I tried to understand the relation between positive cases and the other attributes such as yellow fingers, anxiety, chest pain, chronic disease, fatigue, wheezing, Coughing, short breath, swallowing difficulty, allergy.

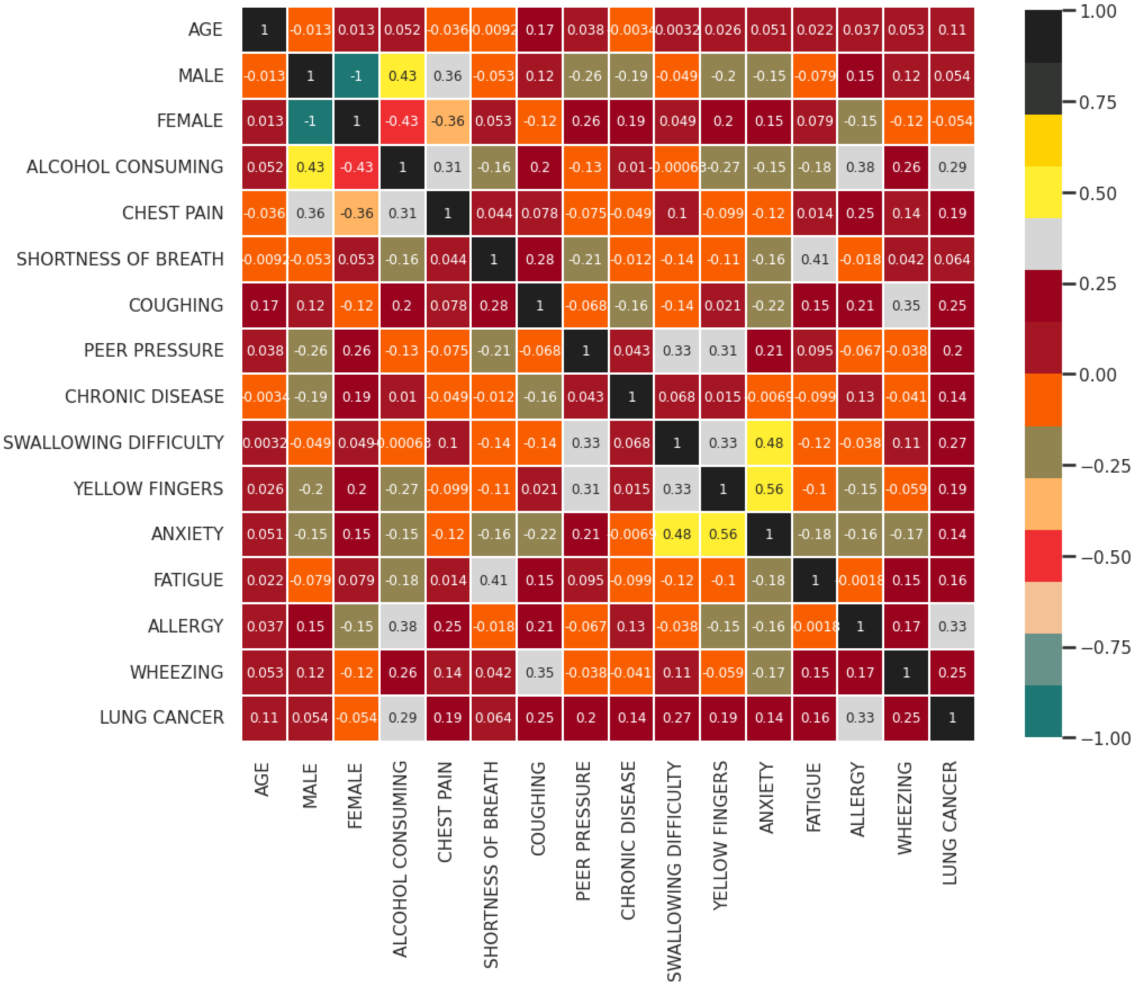
The results for the analysis are as shown in the below graph.

Tool: python, Jupyter Notebook



Finally, I obtained the correlation between each variable using the heatmap() function.

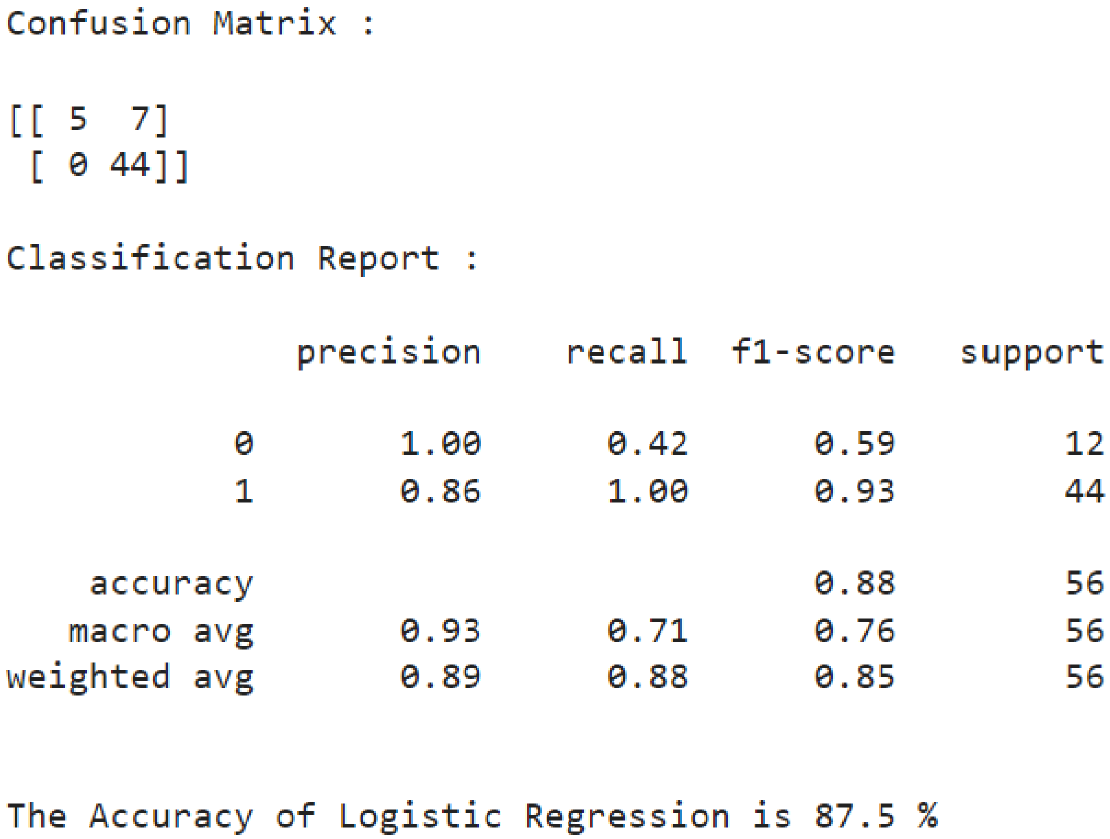
Tool: python, Jupyter Notebook



	AGE	MALE	FEMALE	ALCOHOL CONSUMING	CHEST PAIN	SHORTNESS OF BREATH	COUGHING	PEER PRESSURE	CHRONIC DISEASE	SWALLOWING DIFFICULTY	YELLOW FINGERS	ANXIETY	FATIGUE	ALLERGY	WHEEZING	LUNG CANCER
AGE	1	-0.013	0.013	0.052	-0.036	-0.0092	0.17	0.038	-0.0034	0.0032	0.026	0.051	0.022	0.037	0.053	0.11
MALE	-0.013	1	-1	0.43	0.36	-0.053	0.12	-0.26	-0.19	-0.049	-0.2	-0.15	-0.079	0.15	0.12	0.054
FEMALE	0.013	-1	1	-0.43	-0.36	0.053	-0.12	0.26	0.19	0.049	0.2	0.15	0.079	-0.15	-0.12	-0.054
ALCOHOL CONSUMING	0.052	0.43	-0.43	1	0.31	-0.16	0.2	-0.13	0.01	-0.00063	-0.27	-0.15	-0.18	0.38	0.26	0.29
CHEST PAIN	-0.036	0.36	-0.36	0.31	1	0.044	0.078	-0.075	-0.049	0.1	-0.099	-0.12	0.014	0.25	0.14	0.19
SHORTNESS OF BREATH	-0.0092	-0.053	0.053	-0.16	0.044	1	0.28	-0.21	-0.012	-0.14	-0.11	-0.16	0.41	-0.018	0.042	0.064
COUGHING	0.17	0.12	-0.12	0.2	0.078	0.28	1	-0.068	-0.16	-0.14	0.021	-0.22	0.15	0.21	0.35	0.25
PEER PRESSURE	0.038	-0.26	0.26	-0.13	-0.075	-0.21	-0.068	1	0.043	0.33	0.31	0.21	0.095	-0.067	-0.038	0.2
CHRONIC DISEASE	-0.0034	-0.19	0.19	0.01	-0.049	-0.012	-0.16	0.043	1	0.068	0.015	-0.0069	-0.099	0.13	-0.041	0.14
SWALLOWING DIFFICULTY	0.0032	-0.049	0.049	-0.00063	0.1	-0.14	-0.14	0.33	0.068	1	0.33	0.48	-0.12	-0.038	0.11	0.27
YELLOW FINGERS	0.026	-0.2	0.2	-0.27	-0.099	-0.11	0.021	0.31	0.015	0.33	1	0.56	-0.1	-0.15	-0.059	0.19
ANXIETY	0.051	-0.15	0.15	-0.15	-0.12	-0.16	-0.22	0.21	-0.0069	0.48	0.56	1	-0.18	-0.16	-0.17	0.14
FATIGUE	0.022	-0.079	0.079	-0.18	0.014	0.41	0.15	0.095	-0.099	-0.12	-0.1	-0.18	1	-0.0018	0.15	0.16
ALLERGY	0.037	0.15	-0.15	0.38	0.25	-0.018	0.21	-0.067	0.13	-0.038	-0.15	-0.16	-0.0018	1	0.17	0.33
WHEEZING	0.053	0.12	-0.12	0.26	0.14	0.042	0.35	-0.038	-0.041	0.11	-0.059	-0.17	0.15	0.17	1	0.25
LUNG CANCER	0.11	0.054	-0.054	0.29	0.19	0.064	0.25	0.2	0.14	0.27	0.19	0.14	0.16	0.33	0.25	1

Logistic Regression Model

Logistic regression is a statistical analysis technique used to predict binary outcomes. Yes or no, based on previous observations in the dataset. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

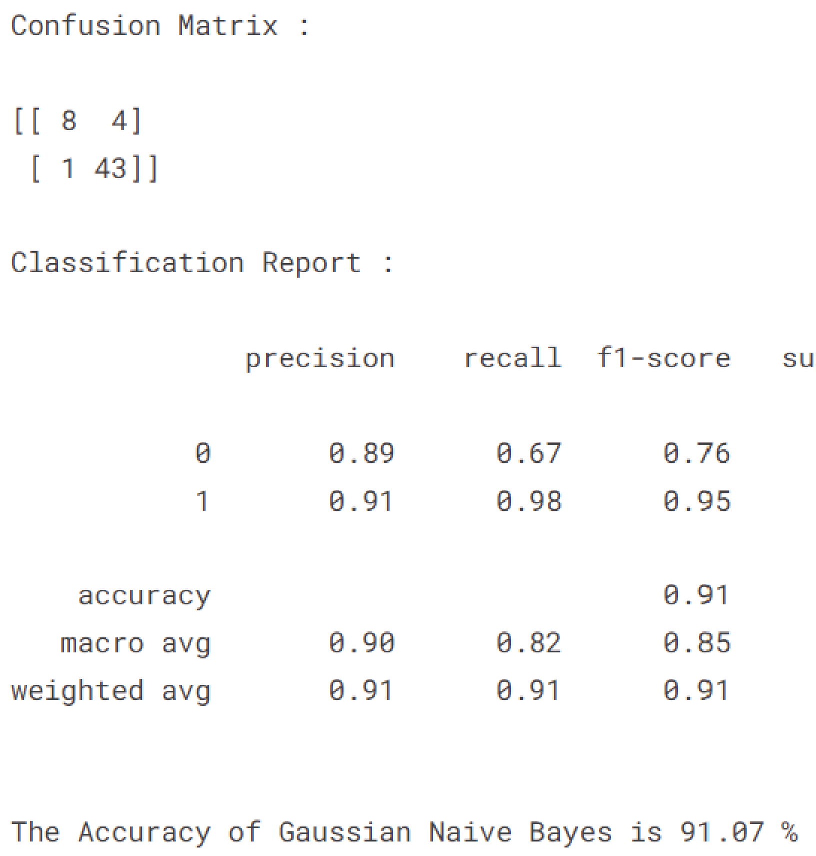


```
Confusion Matrix :

[[ 5  7]
 [ 0 44]]

Classification Report :

              precision    recall  f1-score   support

           0       1.00      0.42      0.59        12
           1       0.86      1.00      0.93        44

    accuracy                           0.88        56
   macro avg       0.93      0.71      0.76        56
weighted avg       0.89      0.88      0.85        56


The Accuracy of Logistic Regression is 87.5 %
```

Tool: python, Jupyter Notebook

Nave Bayes Model

Nave Bayes is a probabilistic machine learning technique that is based on the Bayes theorem and is used for numerous classification purposes. Gaussian Nave Bayes is a naive Bayes generalization. While different functions may be used to estimate data distribution, the Gaussian or normal distribution is the most straightforward to apply since you just need to determine the mean and standard deviation for the training data.

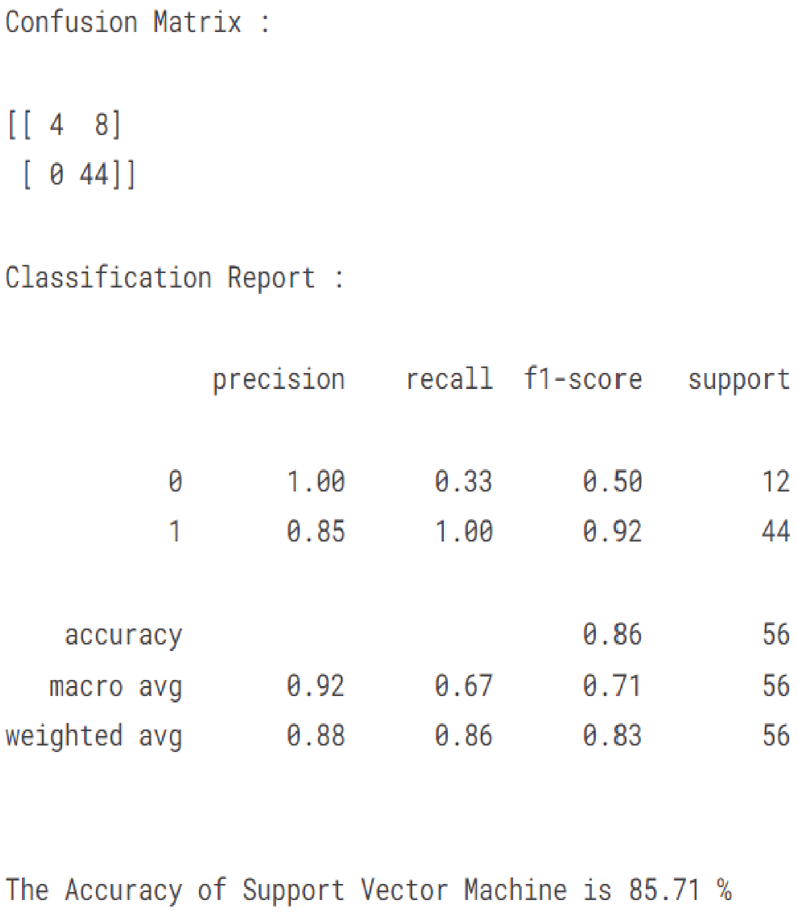


```
Confusion Matrix :

[[ 8  4]
 [ 1 43]]

Classification Report :

              precision    recall  f1-score   support

           0       0.89      0.67      0.76        12
           1       0.91      0.98      0.95        44

    accuracy                           0.91        56
   macro avg       0.90      0.82      0.85        56
weighted avg       0.91      0.91      0.91        56


The Accuracy of Gaussian Naive Bayes is 91.07 %
```

\ Tool: python, Jupyter Notebook

Support Vector Machine Model

A support vector machine (SVM) is a supervised machine learning model that solves two-group classification problems using classification techniques. They can categorize fresh text after providing an SVM model with sets of labeled training data for each category.

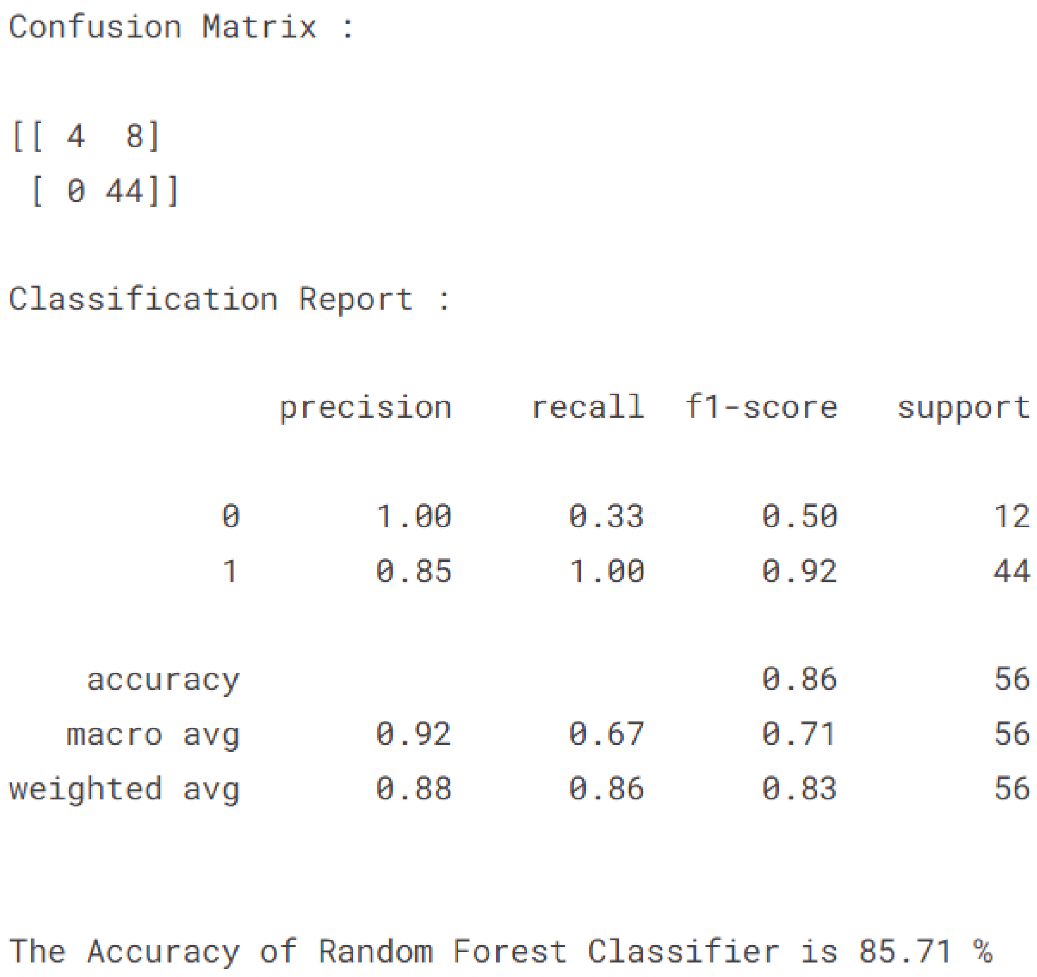


```
Confusion Matrix :

[[ 4  8]
 [ 0 44]]

Classification Report :

              precision    recall  f1-score   support

           0       1.00      0.33      0.50        12
           1       0.85      1.00      0.92        44

    accuracy                           0.86        56
   macro avg       0.92      0.67      0.71        56
weighted avg       0.88      0.86      0.83        56


The Accuracy of Support Vector Machine is 85.71 %
```

Tool: python, Jupyter Notebook

Random Forest Model

A random forest is a machine learning approach for solving regression and classification issues. It makes use of ensemble learning, a technique that combines several classifiers to solve complicated problems.



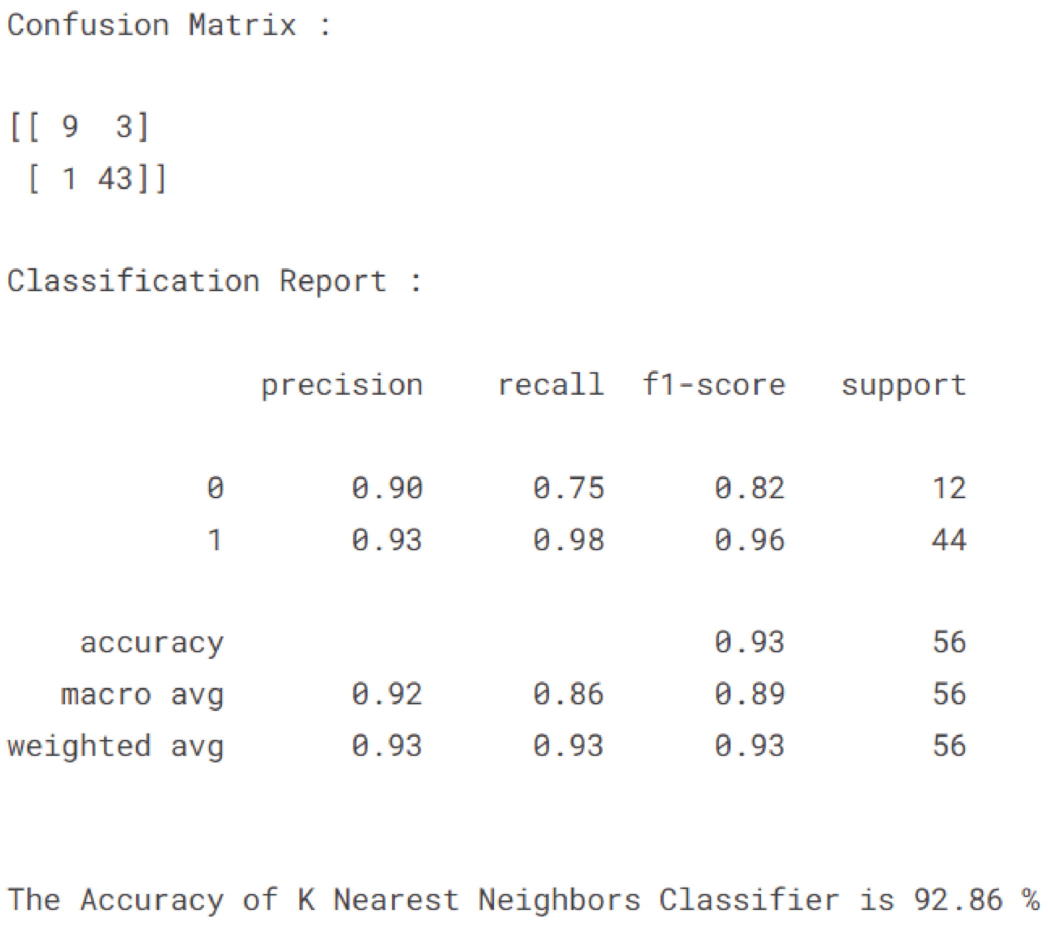
```
Confusion Matrix :

[[ 4  8]
 [ 0 44]]

Classification Report :

              precision    recall  f1-score   support

           0       1.00      0.33      0.50        12
           1       0.85      1.00      0.92        44

    accuracy                           0.86        56
   macro avg       0.92      0.67      0.71        56
weighted avg       0.88      0.86      0.83        56


The Accuracy of Random Forest Classifier is 85.71 %
```

Tool: python, Jupyter Notebook

K-Nearest Neighbours

The k-nearest neighbors (KNN) algorithm is a data categorization approach that estimates the chance that a data point will belong to one of two groups depending on which data points are closest to it.

Tool: python, Jupyter NotebookTool: python, Jupyter Notebook



```
Confusion Matrix :

[[ 9  3]
 [ 1 43]]

Classification Report :

              precision    recall  f1-score   support

           0       0.90      0.75      0.82        12
           1       0.93      0.98      0.96        44

    accuracy                           0.93        56
   macro avg       0.92      0.86      0.89        56
weighted avg       0.93      0.93      0.93        56


The Accuracy of K Nearest Neighbors Classifier is 92.86 %
```

Conclusion

Among all the machine learning models, K Nearest Neighbours performed well with an accuracy of 92.86% and Random Forest performed less with an accuracy of 85.7%.

What I have learnt in this project

This project has a critical role in improving my theoretical and practical knowledge in data mining. I have understood the importance of data preprocessing, feature selection as we have seen a lot of difference in accuracy before and after implementing those techniques. We always had a misconception that if a machine learning model is advanced, then its accuracy would be higher for any type of data. But in reality this is not true as data plays a key role for improving accuracy rather than using an advanced machine learning model. I have also understood the importance of visualization techniques as it would help in understanding the data better, which inturn gives better accuracy.

References

Excel:
Microsoft Corporation. (2018). Microsoft Excel. Retrieved from https://office.microsoft.com/excel

Dataset: Kannan K R. Lung Cancer Dataset 2020. Retrieved from https://www.kaggle.com/datasets/imkrkannan/lung-cancer-dataset-by-staceyinrobert

Python Software Foundation. Python Language Reference, Version 3.10.7. Available to https://www.python.org/downloads/

Kluyver, T. et al., 2016. Jupyter Notebooks – a publishing format for reproducible computational workflows
https://jupyter.org/

Anil Kumar C, Harish S, Ravi P, Svn M, Kumar BPP, Mohanavel V, Alyami NM, Priya SS, Asfaw AK. Lung Cancer Prediction from Text Datasets Using Machine Learning. Biomed Res Int. 2022 Jul 14;2022:6254177. doi: 10.1155/2022/6254177. PMID: 35872862; PMCID: PMC9303121.https://pubmed.ncbi.nlm.nih.gov/35872862/