Lung Cancer Prediction

Course Code : INT 254

Submitted by
Name : Ankit kumar vishwakarma
Registration No : 12008472

In this project(Lung cancer prediction) I have predict the rate of cancer by using machine learning with the use of an excel file.

The mortality rate due to lung cancer is increasing day by day in youths as well as in old persons as compared to other cancers.

Machine learning now days has a great influence to health care sector because of its high computational capability for early prediction of the diseases with accurate data analysis.

What is Lung Cancer Dataset?

- The effectiveness of the cancer prediction system helps people to know their cancer risk at a low cost and
- it also helps the people to take the appropriate decision based on their cancer risk status.
- The data is collected from the website online lung cancer prediction system.

- About Dataset
- The effectiveness of cancer prediction system helps the people to know their cancer risk with low cost and it also helps the people to take the appropriate decision based on their cancer risk status. The data is collected from the website online lung cancer prediction system .
- Total no. of attributes:16

No .of instances:284

Attribute information:

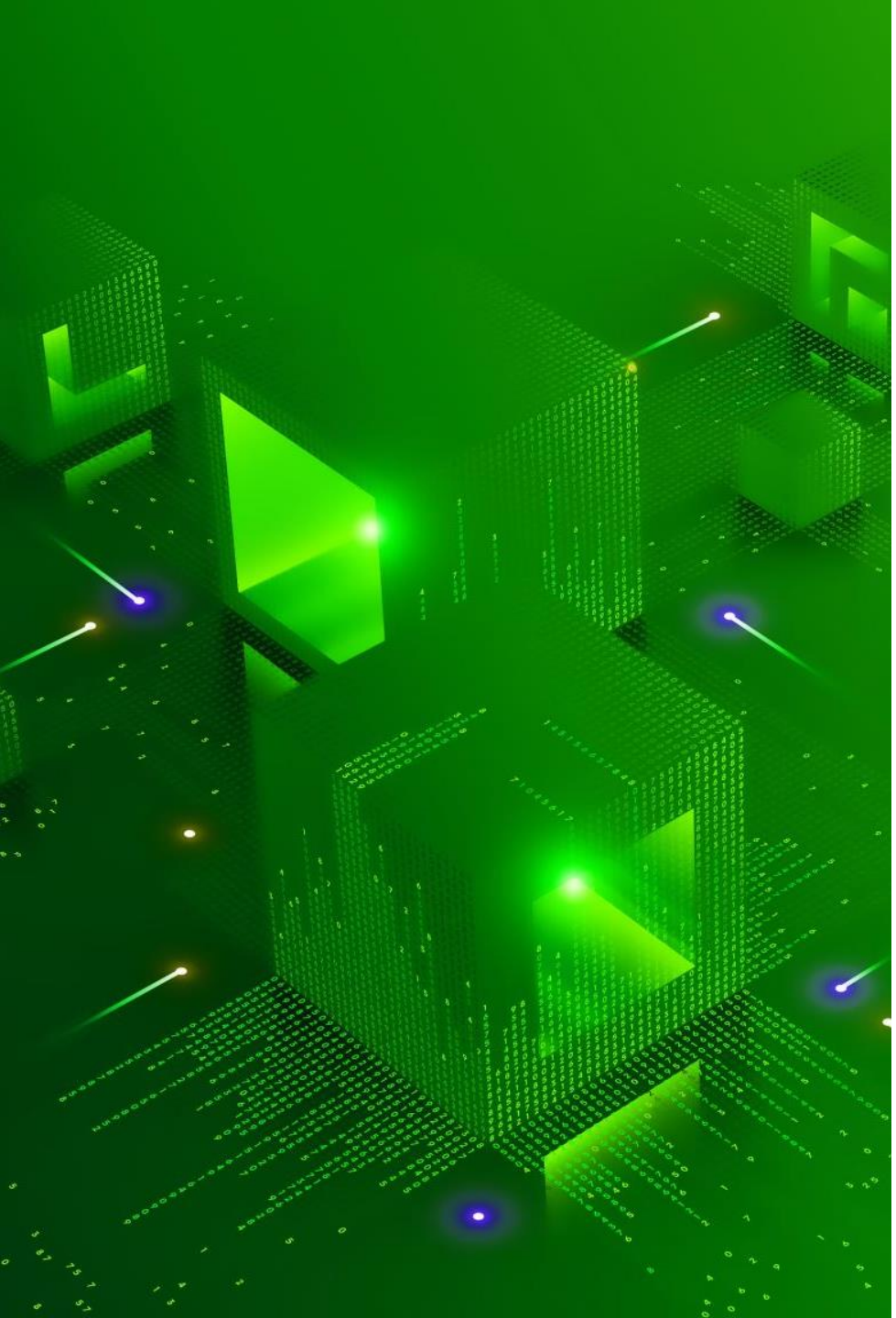
- Gender: M(male), F(female)
- Age: Age of the patient
- Smoking: YES=2 , NO=1.
- Yellow fingers: YES=2 , NO=1.
- Anxiety: YES=2 , NO=1.
- Peer_pressure: YES=2 , NO=1.
- Chronic Disease: YES=2 , NO=1.
- Fatigue: YES=2 , NO=1.
- Allergy: YES=2 , NO=1.
- Wheezing: YES=2 , NO=1.
- Alcohol: YES=2 , NO=1.
- Coughing: YES=2 , NO=1.
- Shortness of Breath: YES=2 , NO=1.
- Swallowing Difficulty: YES=2 , NO=1.
- Chest pain: YES=2 , NO=1.
- Lung Cancer: YES , NO.

Libraries which I have used in this project

- ❑ Pandas
- ❑ Numpy
- ❑ Matplotlib
- ❑ Scikit-learn

Pandas

pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool,built on top of
the Python programming language.
pandas aims to be the fundamental high-level building block for doing practical, real world data analysis in Python.



About pandas library

- A fast and efficient DataFrame object for data manipulation with integrated indexing.
- Tools for reading and writing data between in-memory data structures and different formats: CSV and text files, Microsoft Excel.
- Flexible reshaping and pivoting of data sets.
- Intelligent label-based slicing, fancy indexing, and subsetting of large data sets.
- Columns can be inserted and deleted from data structures for size mutability.
- High performance merging and joining of data sets.

Numpy

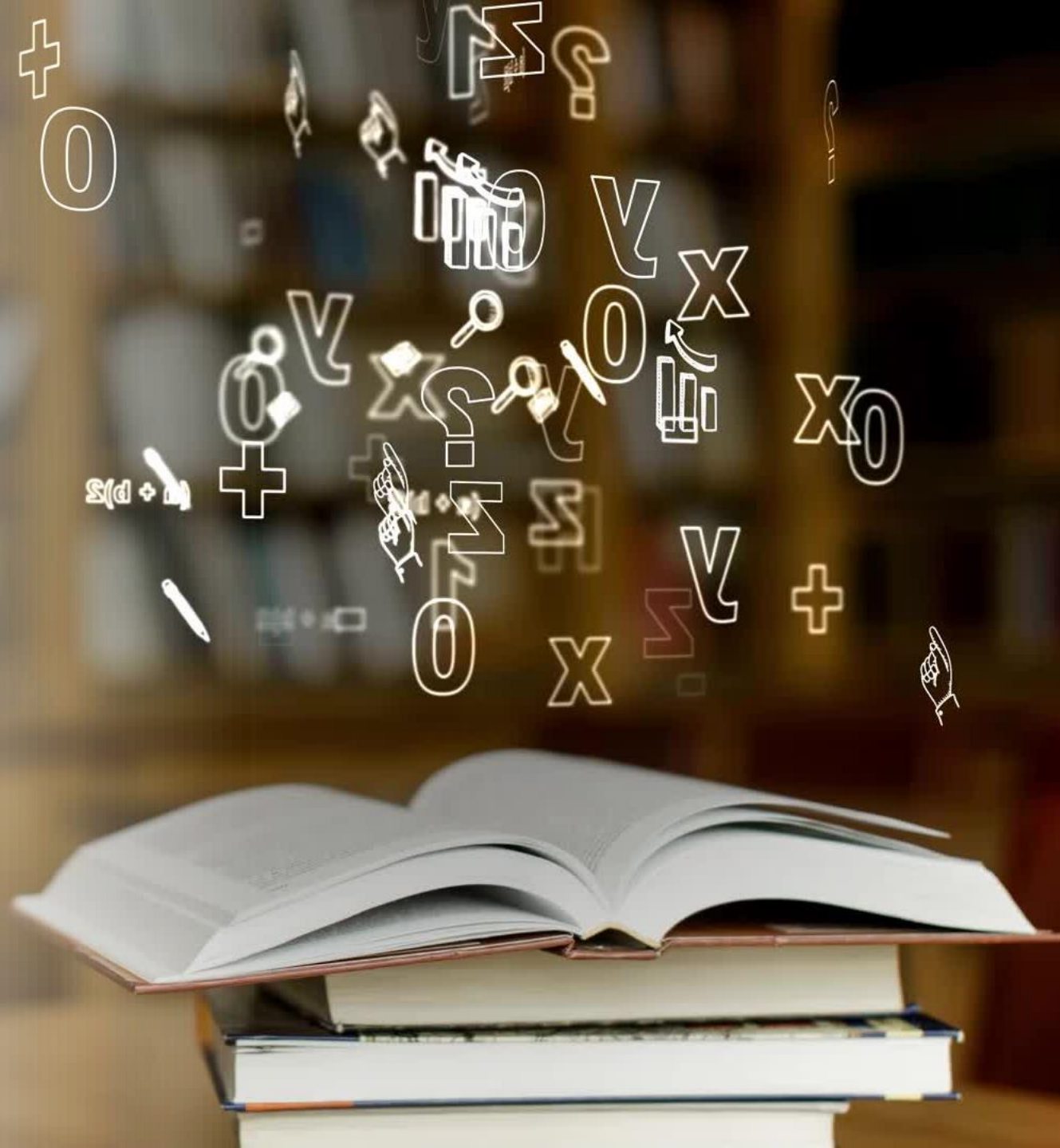
NumPy is the package for computing in Python. It is a Python library that provides a multidimensional array object(such as masked arrays and matrices), and for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

Matplotlib

- Matplotlib is a cross-platform, data visualization and graphical plotting library for Python and its numerical extension NumPy. As such, it offers a viable open source alternative to MATLAB.
- Matplotlib and Pyplot in Python
- The pyplot API has a convenient MATLAB-style stateful interface. In fact, matplotlib was originally written as an open source alternative for MATLAB. The pyplot interface is more commonly used, and is referred to by default in this article.

Scikit-learn

- Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.



```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from pandas.plotting import scatter_matrix
from matplotlib import pyplot
from sklearn.model_selection import train_test_split
""" use of train_test_split
data set - 100 record
split = train and test
  split -  60(to train)
           40(to test )
"""
from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import confusion_matrix
from sklearn.metrics import f1_score
from sklearn.metrics import accuracy_score
from sklearn import tree
import math

print("Datasset:")
dataset = pd.read_csv('lung_cancer.csv')
# print(dataset)
print(len(dataset)) # it wil print lenght of the data
print(dataset.head()) #it will print somedata just to see that what are the datas

scatter_matrix(dataset)
pyplot.show()

A = dataset[dataset.Result == 1] #when the result is 1 then it will taking the vlaue A
B = dataset[dataset.Result == 0] #when the result is o then it will taking the value B
```

```
plt.scatter(A.Age, A.Smokes, color="Black", label="1", alpha=0.4)  #1 is for who is having cancer
plt.scatter(B.Age, B.Smokes, color="Blue", label="0", alpha=0.4)   #0 is for who is having no cancer
plt.xlabel("Age")
plt.ylabel("Smokes")
plt.legend()
plt.title("Smokes vs Age")
plt.show()


plt.scatter(A.Age, A.Alkhol, color="Black", label="1",alpha=0.4)
plt.scatter(B.Age, B.Alkhol, color="Blue", label="0",alpha=0.4)
plt.xlabel("Age")
plt.ylabel("Alkhol")
plt.legend()
plt.title("Alkhol vs Age")
plt.show()

plt.scatter(A.Smokes, A.Alkhol, color="Black", label="1",alpha=0.4)
plt.scatter(B.Smokes, B.Alkhol, color="Blue", label="0",alpha=0.4)
plt.xlabel("Smokes")
plt.ylabel("Alkhol")
plt.legend()
plt.title("Alkhol vs Smokes")
plt.show()


#spliting dataset
x = dataset.iloc[:,3:5]
y = dataset.iloc[:,6]
x_train, x_test, y_train, y_test = train_test_split(x,y, random_state=0, test_size=0.2)


# feature Scaling

sc_x = StandardScaler()
x_train = sc_x.fit_transform(x_train) #fitting the model
x_test = sc_x.transform(x_test)
```

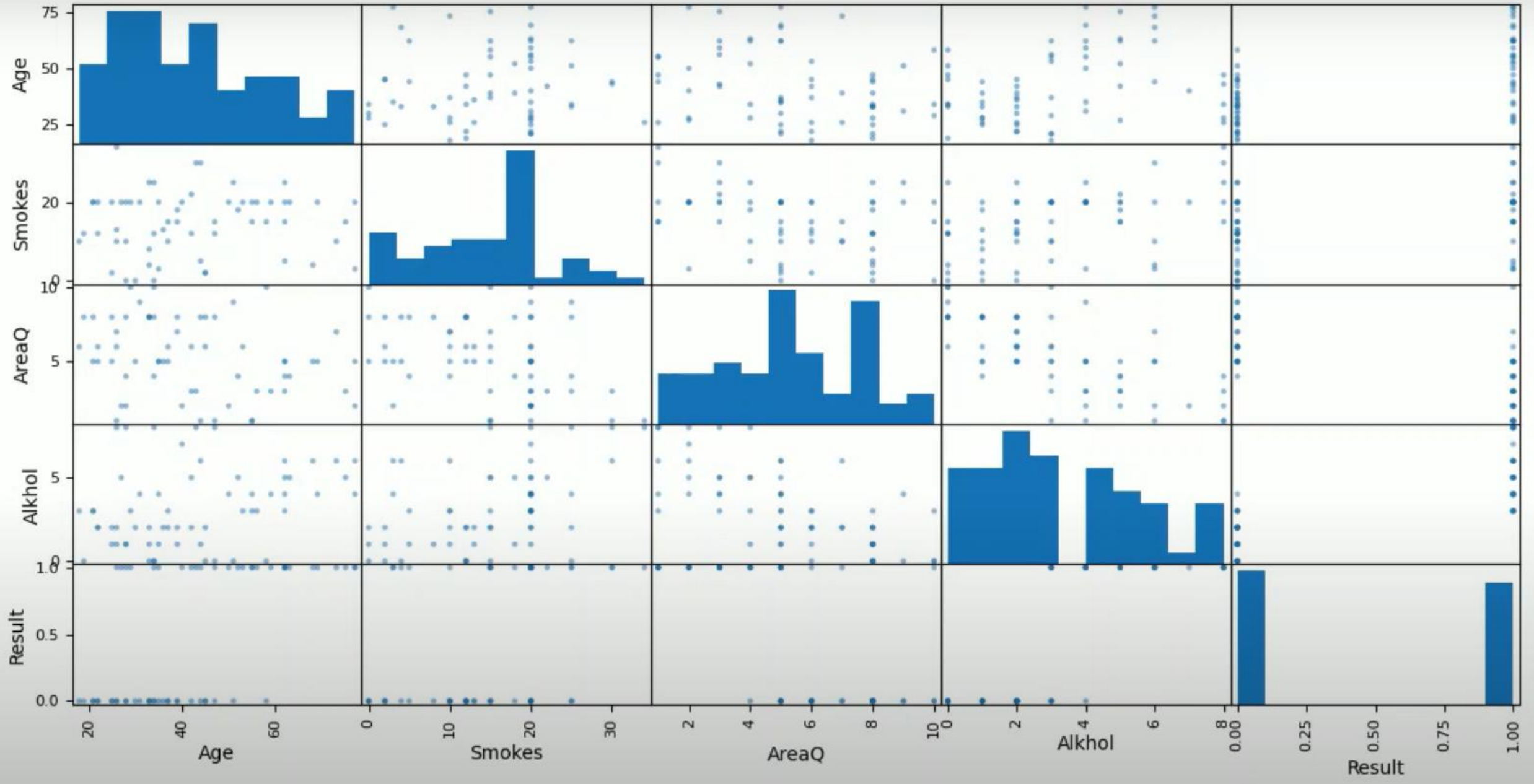
```python
print('----------------------****Using KNN Algorithm***-----------------------------')
a = math.sqrt(len(y_train))
print(a)


#defining a model - KNN
classifier = KNeighborsClassifier(n_neighbors=5, p=2, metric = 'eculidien')

#fit model
classifier.fit(x_train, y_train)

#predict the test result
y_pred = classifier.predict(x_test)
print(y_pred)


#Evaluate model
#confusion matrix
cm = confusion_matrix(y_test, y_pred)
print("Confusion Matrix")
print(cm)
print("In Confusion Matrix:-----")
print("Position 1.1 shows the patients that don't have Cancer, In this case = 8")
print("Position 1.2 shows the number of patients that have higher risk of Cancer, In this case = 1")
print("Postion 2.1 shows the Incorrect Value, In this case = 2")
print("Position 2.2 shows the correct number of patients that have Cancer, In this case = 2")

print('F1 Score : ',(f1_score(y_test, y_pred))*100)
print('ACCURACY :',(accuracy_score(y_test, y_pred))*100)
```

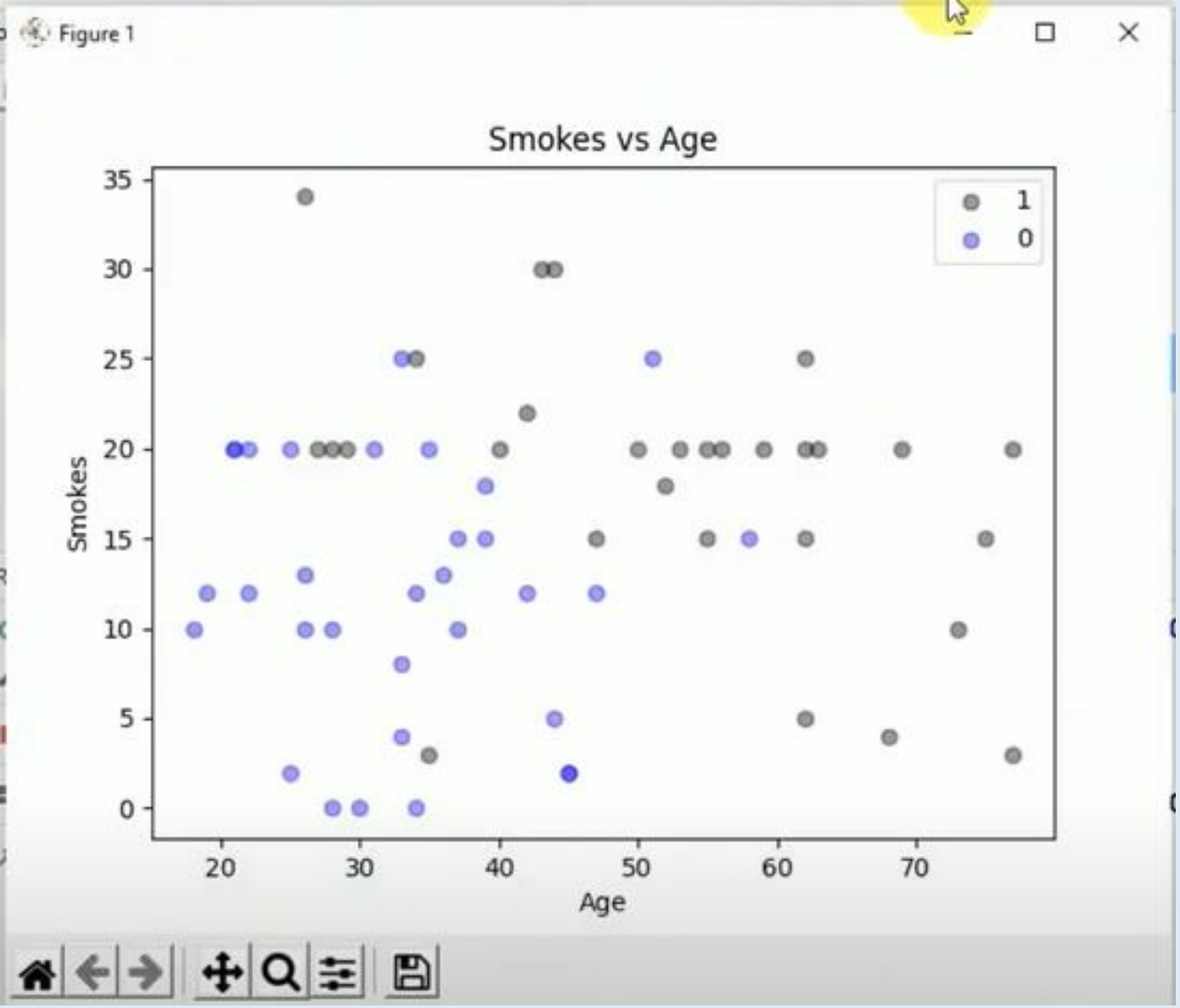
```
#using decision treee
print('----------------------****Using Decision Tree Algorithm***----------------------------')
c=tree.DecisionTreeClassifier()
c.fit(x_train, y_train)
accu_train = np.sum(c.predict(x_train)==y_train) / float(y_train.size)
accu_test = np.sum(c.predict(x_test)==y_test) / float(y_test.size)
print('Classsification accuracy on train',(accu_train)*100)
print('Classification accuracy on test', (accu_test)*100)
```

Screenshots of the project

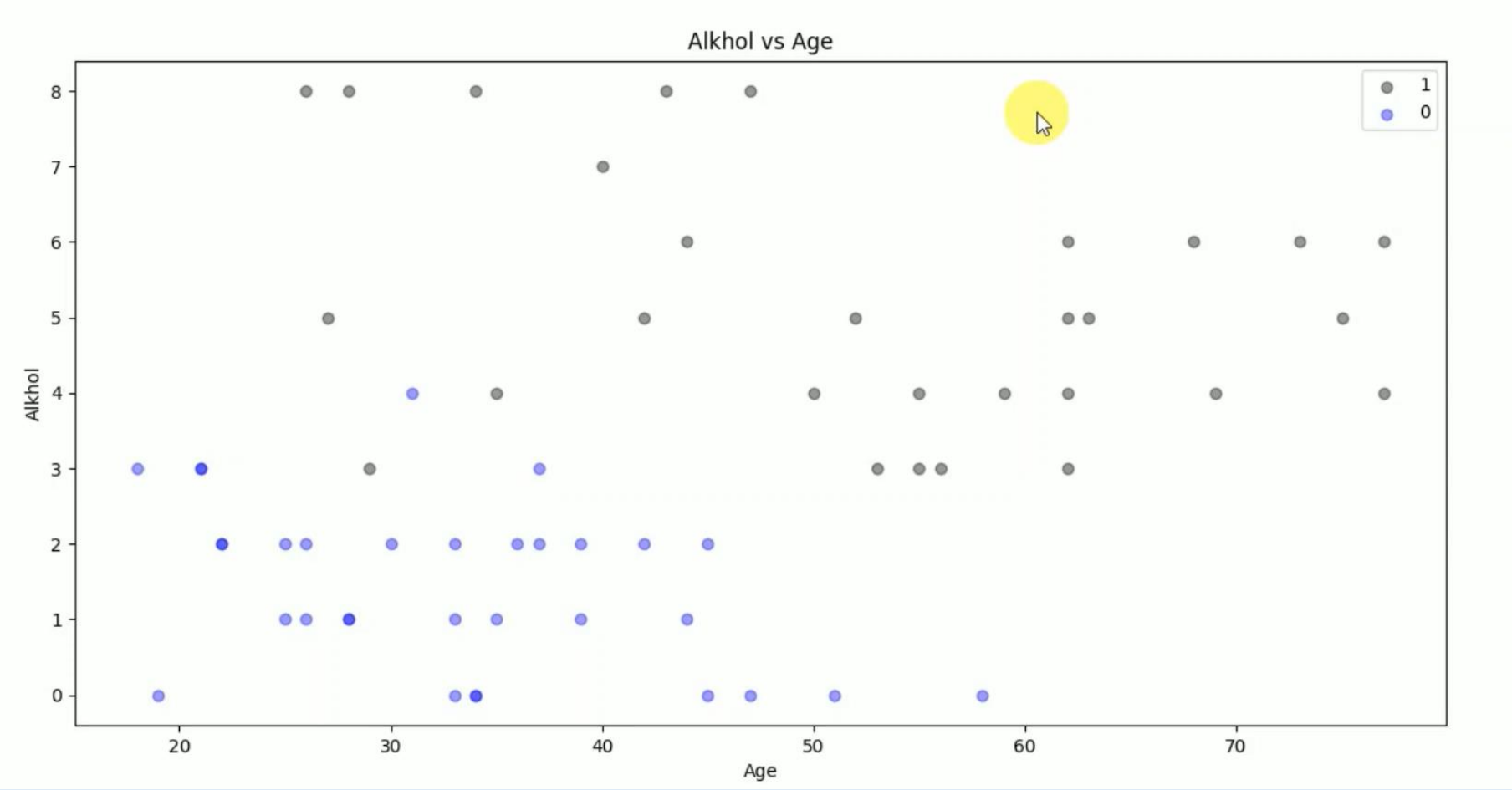
	Name	Surname	Age	Smokes	AreaQ	Alkhol	Result
0	John	Wick	35	3	5	4	1
1	John	Constantine	27	20	2	5	1
2	Camela	Anderson	30	0	5	2	0
3	Alex	Telles	28	0	8	1	0
4	Diego	Maradona	68	4	5	6	1
5	Cristiano	Ronaldo	34	0	10	0	0
6	Mihail	Tal	58	15	10	0	0
7	Kathy	Bates	22	12	5	2	0
8	Nicole	Kidman	45	2	6	0	0
9	Ray	Milland	52	18	4	5	1
10	Fredric	March	33	4	8	0	0
11	Yul	Brynner	18	10	6	3	0
12	Joan	Crawford	25	2	5	1	0
13	Jane	Wyman	28	20	2	8	1
14	Anna	Magnani	34	25	4	8	1
15	Katharine	Hepburn	39	18	8	1	0
16	Katharine	Hepburn	42	22	3	5	1
17	Barbra	Streisand	19	12	8	0	0
18	Maggie	Smith	62	5	4	3	1
19	Glenda	Jackson	73	10	7	6	1
20	Jane	Fonda	55	15	1	3	1
21	Maximilian	Schell	33	8	8	1	0
22	Gregory	Peck	22	20	6	2	0
45	Gwyneth	Paltrow	21	20	8	3	0
46	Halle	Berry	31	20	9	4	0
47	Nicole	Kidman	28	10	4	1	0
48	Charlize	Theron	53	20	6	3	1
49	Katharine	Hepburn	62	20	5	6	1
50	Katharine	Hepburn	42	12	6	2	0
51	Barbra	Streisand	44	30	1	6	1
52	Maggie	Smith	26	34	1	8	1
53	Glenda	Jackson	35	20	5	1	0
54	Ernest	Borgnine	26	13	6	1	0
55	Alec	Guinness	77	20	5	4	1
56	Charlton	Heston	75	15	3	5	1
57	Gregory	Peck	43	30	3	8	1
58	Sidney	Poitier	51	25	9	0	0

- These are the elements which are sotred in the csv file those are the all the Names, Surame, Age, Smokes, AreaQ, Akhol and the Results(0,1).
- 1 for who is having cancer and 0 for who is having no lung cancer.

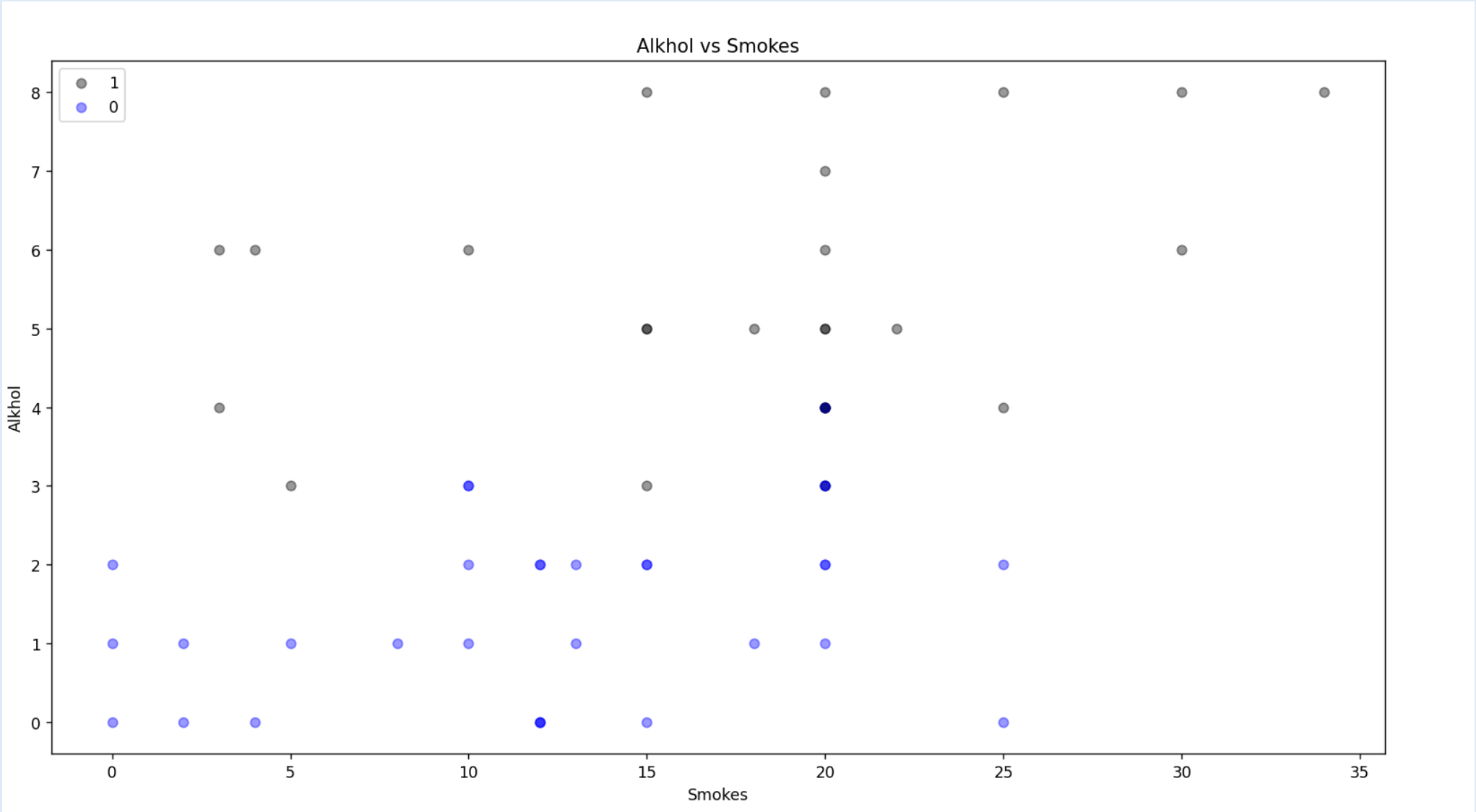

Age
Smokes
AreaQ
Alkhol
Result
75
50
25
20
0
10
5
0
5
0
1.0
0.5
0.0
20
40
60
0
10
20
30
2
4
6
8
10
0
2
4
6
8
0.00
0.25
0.50
0.75
1.00



- This is the next graph between smokes vs Age by the use of legend. Legend which is holding the marker and the value 0 and 1. Where 1 is for black and blue is for 0.
- Black is for those persons who are having cancer and blue is for who are having not cancer.



- This is the next graph between Alkhol vs Age by the use of legend. Legend which is holding the marker and the value 0 and 1. Where 1 is for black and blue is for 0.
- Black is for those persons who are having cancer and blue is for who are having not cancer.


Alkhol vs Smokes
1
0
Alkhol
Smokes
0
1
2
3
4
5
6
7
8
0
5
10
15
20
25
30
35

```
----------------****Using KNN Algorithm****---------------
6.928203230275509
[0 0 0 1 0 0 0 0 1 1 0 0 0]
Confusion Matrix:
[[8 1]
 [2 2]]
In Confusion Matrix:-----
Position 1.1 shows the patients that don't have Cancer, In this case = 8
Position 1.2 shows the number of patients that have higher risk of Cancer, In this case = 1
Position 2.1 shows the Incorrect Value, In this case = 2
Position 2.2 shows the correct number of patients that have Cancer, In this case = 2
F1 Score :  57.14285714285715
ACCURACY :  76.92307692307693
```

```
---------------****Using Decision Tree Algorithm****--------------
Classification accuracy  on train 95.83333333333334
Classification accuracy  on test 69.23076923076923
```

Thank you