Name : Aryan Sirdesai; Roll No.: TAO20175;

Problem Statement : Data Analytics III

Implement Simple Naïve Bayes classification algorithm using Python/R on iris.csv dataset. Compute Confusion matrix to find TP, FP, TN, FN, Accuracy, Error rate, Precision, Recall on the given dataset.

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

In [2]:
```
iris = pd.read_csv('Iris.csv')
iris
```

Out[2]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...	...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [3]:
```
iris.head()
```

Out[3]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

In [4]:
```
iris.describe()
```

Out[4]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

In [5]:

```
iris.isnull().sum()
```

Out[5]:

```
Id               0
SepalLengthCm    0
SepalWidthCm     0
PetalLengthCm    0
PetalWidthCm     0
Species          0
dtype: int64
```

In [6]:

```
iris.columns
```

Out[6]:

```
Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm',
       'Species'],
      dtype='object')
```

In [7]:

```
X = iris.iloc[:,:4].values
Y = iris['Species'].values
```

In [8]:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.2)
```

In [9]:

```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

In [10]:

```
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train, y_train)
```

Out[10]:

```
GaussianNB()
```

In [11]:

```
y_pred = classifier.predict(X_test)
y_pred
```

Out[11]:

```
array(['Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-versicolor',
       'Iris-virginica', 'Iris-versicolor', 'Iris-virginica',
       'Iris-virginica', 'Iris-setosa', 'Iris-virginica', 'Iris-setosa',
       'Iris-versicolor', 'Iris-setosa', 'Iris-virginica', 'Iris-setosa',
       'Iris-setosa', 'Iris-setosa', 'Iris-setosa', 'Iris-versicolor',
       'Iris-virginica', 'Iris-setosa', 'Iris-versicolor',
       'Iris-virginica', 'Iris-setosa', 'Iris-versicolor', 'Iris-setosa',
       'Iris-versicolor', 'Iris-setosa', 'Iris-setosa', 'Iris-versicolor'],
      dtype='<U15')
```

In [12]:

```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
cm
```

Out[12]:

```
array([[15,  0,  0],
       [ 0,  8,  0],
       [ 0,  0,  7]], dtype=int64)
```

In [13]:

```
from sklearn.metrics import accuracy_score
print ("Accuracy : ", accuracy_score(y_test, y_pred))
cm
```

Out[13]:

```
Accuracy :  1.0
array([[15,  0,  0],
       [ 0,  8,  0],
       [ 0,  0,  7]], dtype=int64)
```

In [14]:

```
df = pd.DataFrame({'Real Values':y_test, 'Predicted Values':y_pred})
df
```

Out[14]:

	Real Values	Predicted Values
0	Iris-setosa	Iris-setosa
1	Iris-setosa	Iris-setosa
2	Iris-setosa	Iris-setosa
3	Iris-versicolor	Iris-versicolor
4	Iris-virginica	Iris-virginica
5	Iris-versicolor	Iris-versicolor
6	Iris-virginica	Iris-virginica
7	Iris-virginica	Iris-virginica
8	Iris-setosa	Iris-setosa
9	Iris-virginica	Iris-virginica
10	Iris-setosa	Iris-setosa
11	Iris-versicolor	Iris-versicolor
12	Iris-setosa	Iris-setosa
13	Iris-virginica	Iris-virginica
14	Iris-setosa	Iris-setosa
15	Iris-setosa	Iris-setosa
16	Iris-setosa	Iris-setosa
17	Iris-setosa	Iris-setosa
18	Iris-versicolor	Iris-versicolor
19	Iris-virginica	Iris-virginica
20	Iris-setosa	Iris-setosa
21	Iris-versicolor	Iris-versicolor
22	Iris-virginica	Iris-virginica
23	Iris-setosa	Iris-setosa
24	Iris-versicolor	Iris-versicolor
25	Iris-setosa	Iris-setosa
26	Iris-versicolor	Iris-versicolor
27	Iris-setosa	Iris-setosa
28	Iris-setosa	Iris-setosa
29	Iris-versicolor	Iris-versicolor

In [20]:

```
iris.iloc[1:3,2:4]
```

Out[20]:

	SepalWidthCm	PetalLengthCm
1	3.0	1.4
2	3.2	1.3

```
In [ ]:
```