Experiment - 6

Student Name: Aniket Kumar
UID: 20BCS5306
Branch: CSE
Section/Group: 20BCS_WM-703/B
Semester: 5th
Date of Performance: 21/10/2022
Subject Name: Machine Learning
Subject Code: 20CSP-317

1. Aim:
Implement Naïve Bayesonany dataset.

2. Objective:
Gaussian Naïve Bayes is used when we assume all the continuous variables associated with each feature to be distributed according to Gaussian Distribution. Gaussian Distribution is also called Normal distribution.

3. Result and Output:

```
In [1]: from sklearn.datasets import load_iris

In [2]: from sklearn.model_selection import train_test_split
        from sklearn.naive_bayes import GaussianNB

In [3]: x,y = load_iris(return_X_y= True)

In [4]: x

Out[4]: array([[5.1, 3.5, 1.4, 0.2],
               [4.9, 3. , 1.4, 0.2],
               [4.7, 3.2, 1.3, 0.2],
               [4.6, 3.1, 1.5, 0.2],
               [5. , 3.6, 1.4, 0.2],
               [5.4, 3.9, 1.7, 0.4],
               [4.6, 3.4, 1.4, 0.3],
               [5. , 3.4, 1.5, 0.2],
               [4.4, 2.9, 1.4, 0.2],
               [4.9, 3.1, 1.5, 0.1],
               [5.4, 3.7, 1.5, 0.2],
               [4.8, 3.4, 1.6, 0.2],
               [4.8, 3. , 1.4, 0.1],
               [4.3, 3. , 1.1, 0.1],
               [5.8, 4. , 1.2, 0.2],
               [5.7, 4.4, 1.5, 0.4],
               [5.4, 3.9, 1.3, 0.4],
               [5.1, 3.5, 1.4, 0.3],
               [5.7, 3.8, 1.7, 0.3],
```

In [5]:
```
y
```

Out[5]:
```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2])
```

In [6]:
```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=1)
```

In [7]:
```
nb = GaussianNB()
```

In [8]:
```
nb.fit(x_train,y_train)
```

Out[8]:
```
GaussianNB()
```

In [9]:
```
y_pre = nb.predict(x_test)
```

In [10]:
```
from sklearn.metrics import accuracy_score
accuracy_score(y_pre,y_test)
```

Out[10]:
```
0.9666666666666667
```

In [11]:
```
from sklearn.metrics import confusion_matrix
cm=confusion_matrix(y_test,y_pre)
```

In [12]:
```
cm
```

Out[12]:
```
array([[11,  0,  0],
       [ 0, 12,  1],
       [ 0,  0,  6]], dtype=int64)
```