

Program 6: Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.

```
import pandas as pd

#reading the dataset

data=pd.read_csv('data.csv')

#calculating the total no.,no. of positive and no. of negative instances

te=len(data)

np=len(data.loc[data[data.columns[-1]]=='Yes'])

nn=te-np

#dividing the dataset into training and test

training=data.sample(frac=0.75,replace=False)

test=pd.concat([data, training, training]).drop_duplicates(keep=False)

print("Training Set : \n",training)

print("\nTest Data Set : \n",test)

#For every value of each attribute calculate the negative and positive probability

prob={}

for col in training.columns[:-1]:

    prob[col]={ }

    vals=set(data[col])

    for val in vals:

        temp=training.loc[training[col]==val]

        pe=len(temp.loc[temp[temp.columns[-1]]=='Yes'])

        ne=len(temp)-pe

        prob[col][val]=[pe/np,ne/nn]

#Using Bayes Theorem to Predict the output
```

```

prediction=[]
right_prediction=0
for i in range(len(test)):
    row=test.iloc[i,:]
    fpp=np/te
    fpn=nn/te
    for col in test.columns[:-1]:
        fpp*=prob[col][row[col]][0]
        fpn*=prob[col][row[col]][1]
    if fpp>fpn:
        prediction.append('Yes')
    else:
        prediction.append('No')
    if prediction[-1]==row[-1]:
        right_prediction+=1
#output
print("\nActual Values : ',list(test[test.columns[-1]]))
print('Predicted : ',prediction)
print('Accuracy : ',right_prediction/len(test))

```