KIET Group of Institutions, Ghaziabad

Department of Computer Applications

(An ISO – 9001: 2015 Certified & 'A' Grade accredited Institution by NAAC)

Artificial Intelligence Lab

KCA 351: Session 2021-22

Experiment – No-10

Problem Statement : **Given an example of weather conditions and playing sports. You need to calculate the probability of playing sports. Now, you need to classify whether players will play or not, based on the weather condition.**

Whether	Play
Sunny	No
Sunny	No
Overcast	Yes
Rainy	Yes
Rainy	Yes
Rainy	No
Overcast	Yes
Sunny	No
Sunny	Yes
Rainy	Yes
Sunny	Yes
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table

Whether	No	Yes
Overcast		4
Sunny	2	3
Rainy	3	2
Total	5	9

Likelihood Table 1

Whether	No	Yes		
Overcast		4	=4/14	0.29
Sunny	2	3	=5/14	0.36
Rainy	3	2	=5/14	0.36
Total	5	9		
	=5/14	=9/14		
	0.36	0.64		

Likelihood Table 2

Whether	No	Yes	Posterior Probability for No	Posterior Probability for Yes
Overcast		4	0/5=0	4/9=0.44
Sunny	2	3	2/5=0.4	3/9=0.33
Rainy	3	2	3/5=0.6	2/9=0.22
Total	5	9		

Program:

```
weather =['sunny','sunny','overcast','rainy','rainy','rainy','overcast','s
unny','rainy','sunny','overcast','overcast','rainy']
temp = ['hot','hot','hot','mild','cool','cool','cool','mild','cool','mild'
,'mild','hot','hot']
play = ['no','no','yes','yes','no','yes','no','yes','yes','no','yes','no',
'no']
from sklearn import preprocessing
le = preprocessing.LabelEncoder()
weather_encoded = le.fit_transform(weather)
print(weather_encoded)
type(weather_encoded)
temp_encoded = le.fit_transform(temp)
play_encoded = le.fit_transform(play)
print(temp_encoded)
print(play_encoded)
features = list(zip(weather_encoded,temp_encoded))
print(features)
from sklearn.naive_bayes import GaussianNB
model = GaussianNB()
model.fit(features,play_encoded)
predicted = model.predict([[0,2]])
print("Predicted Value = ",predicted)
```

Output :

```
[2 2 0 1 1 1 0 2 2 1 2 0 0 1]

Temp: [1 1 1 2 0 0 0 2 0 2 2 2 1 2]


Play: [0 0 1 1 1 0 1 0 1 1 1 1 1 0]

[(2, 1), (2, 1), (0, 1), (1, 2), (1, 0), (1, 0), (0, 0), (2, 2), (2, 0), (1,
2), (2, 2), (0, 2), (0, 1), (1, 2)]

Predicted Value: [1]
```