|  |
| --- |
| **Name: Vyshnavi Babu S**  **Roll No:55**  **Batch: MCA-B**  **Date: 17-10-2022** |

**DATA SCIENCE LAB**

**Experiment No.: 9**

**Aim**

Naïve bayes using golf dataset with confusion matrix

**Procedure**

import pandas as pd import numpy as np from sklearn import preprocessing

data= pd.read\_csv('golf-dataset.csv') data

Output

OutlookTemp Humidity Windy Play Golf

1. Rainy Hot High False No
2. Rainy Hot High True No
3. Overcast Hot High False Yes
4. Sunny Mild High False Yes
5. Sunny Cool Normal False Yes
6. Sunny Cool Normal True No
7. Overcast Cool Normal True Yes
8. Rainy Mild High False No
9. Rainy Cool Normal False Yes
10. Sunny Mild Normal False Yes
11. Rainy Mild Normal True Yes
12. Overcast Mild High True Yes
13. Overcast Hot Normal False Yes
14. Sunny Mild High True No

label\_encoder = preprocessing.LabelEncoder() data['Outlook']= label\_encoder.fit\_transform(data['Outlook']) data['Temp']= label\_encoder.fit\_transform(data['Temp']) data['Humidity']= label\_encoder.fit\_transform(data['Humidity']) data['Windy']= label\_encoder.fit\_transform(data['Windy']) data['Play Golf']= label\_encoder.fit\_transform(data['Play Golf'])

**Outlook Temp Humidity Windy Play Golf**

1. 1 1 0 0 0
2. 1 1 0 1 0
3. 0 1 0 0 1
4. 2 2 0 0 1
5. 2 0 1 0 1
6. 2 0 1 1 0
7. 0 0 1 1 1
8. 1 2 0 0 0
9. 1 0 1 0 1
10. 2 2 1 0 1
11. 1 2 1 1 1
12. 0 2 0 1 1
13. 0 1 1 0 1
14. 2 2 0 1 0

X=data.drop('Play Golf', axis=1)

X

OutlookTemp Humidity Windy

1. 1 1 0 0
2. 1 1 0 1
3. 0 1 0 0
4. 2 2 0 0
5. 2 0 1 0
6. 2 0 1 1
7. 0 0 1 1
8. 1 2 0 0
9. 1 0 1 0
10. 2 2 1 0
11. 1 2 1 1

1. 0 2 0 1
2. 0 1 1 0
3. 2 2 0 1

X=data.drop('Play Golf', axis=1)

Y=data['Play Golf']

Y

1. 0
2. 0
3. 1
4. 1
5. 1
6. 0
7. 1
8. 0
9. 1
10. 1
11. 1
12. 1
13. 1
14. 0

Name: Play Golf, dtype: int64

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, Y, test\_size=0.1, ra ndom\_state=42)

from sklearn.naive\_bayes import GaussianNB gnb = GaussianNB() gnb.fit(X\_train, y\_train)

y\_pred = gnb.predict(X\_test)

from sklearn import metrics print("Gaussian Naive Bayes model accuracy(in %):", metrics.accuracy\_score( y\_test, y\_pred)\*100)

from sklearn import metrics confusion\_matrix = metrics.confusion\_matrix(Y, Y)

confusion\_matrix

# Output

array([[5, 0], [0, 9]])