MRIDUL HARISH, CED18I034, PROBLEM SET 7

Q1 - Implement the Naïve Bayes Classifier on the below given dataset. Test record for the given dataset is (Rainy, Cool, Normal, True). Also test the same on a large dataset with a sample test record.

In [ ]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.naive_bayes import GaussianNB
```

In [ ]:
```
golf_df=pd.read_csv("golf-dataset.csv")
golf_df
```

Out[ ]:

	Outlook	Temp	Humidity	Windy	Play Golf
0	Rainy	Hot	High	False	No
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	Yes
5	Sunny	Cool	Normal	True	No
6	Overcast	Cool	Normal	True	Yes
7	Rainy	Mild	High	False	No
8	Rainy	Cool	Normal	False	Yes
9	Sunny	Mild	Normal	False	Yes
10	Rainy	Mild	Normal	True	Yes
11	Overcast	Mild	High	True	Yes
12	Overcast	Hot	Normal	False	Yes
13	Sunny	Mild	High	True	No

In [ ]:
```
golf_df.loc[len(golf_df)]=['Rainy','Cool','Normal',True,'No']
```

In [ ]:
```
train_x=golf_df.iloc[:,[0,1,2,3]].values
train_y=golf_df.iloc[:,-1].values
```

In [ ]:
```
train_x
```

Out[ ]:
```
array([['Rainy', 'Hot', 'High', False],
       ['Rainy', 'Hot', 'High', True],
       ['Overcast', 'Hot', 'High', False],
```

```
       ['Sunny', 'Mild', 'High', False],
       ['Sunny', 'Cool', 'Normal', False],
       ['Sunny', 'Cool', 'Normal', True],
       ['Overcast', 'Cool', 'Normal', True],
       ['Rainy', 'Mild', 'High', False],
       ['Rainy', 'Cool', 'Normal', False],
       ['Sunny', 'Mild', 'Normal', False],
       ['Rainy', 'Mild', 'Normal', True],
       ['Overcast', 'Mild', 'High', True],
       ['Overcast', 'Hot', 'Normal', False],
       ['Sunny', 'Mild', 'High', True],
       ['Rainy', 'Cool', 'Normal', True]], dtype=object)
```

In [ ]:
```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
for i in range(len(train_x[0])):
    train_x[:,i] = le.fit_transform(train_x[:,i])
```

Removing the last row from both data and label before training the dataset

In [ ]:
```
test_x=train_x[-1]
test_x
```

Out[ ]:
```
array([1, 0, 1, 1], dtype=object)
```

In [ ]:
```
train_x=train_x[:-1]
train_x
```

Out[ ]:
```
array([[1, 1, 0, 0],
       [1, 1, 0, 1],
       [0, 1, 0, 0],
       [2, 2, 0, 0],
       [2, 0, 1, 0],
       [2, 0, 1, 1],
       [0, 0, 1, 1],
       [1, 2, 0, 0],
       [1, 0, 1, 0],
       [2, 2, 1, 0],
       [1, 2, 1, 1],
       [0, 2, 0, 1],
       [0, 1, 1, 0],
       [2, 2, 0, 1]], dtype=object)
```

In [ ]:
```
train_y=train_y[:-1]
```

In [ ]:
```
classifier = GaussianNB()
classifier.fit(train_x, train_y)
```

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

In [ ]:
```
pred_y=classifier.predict([test_x])
```

```
C:\Users\hp\anaconda3\lib\site-packages\sklearn\base.py:566: FutureWarning: Arrays of
bytes/strings is being converted to decimal numbers if dtype='numeric'. This behavior
```