

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import LabelEncoder
```

```
data_train = pd.read_csv("/content/play_tennis_train.csv")
data_train.head()
```

	day	outlook	temp	humidity	wind	play
0	D1	Sunny	Hot	High	Weak	No
1	D2	Sunny	Hot	High	Strong	No
2	D3	Overcast	Hot	High	Weak	Yes
3	D4	Rain	Mild	High	Weak	Yes
4	D5	Rain	Cool	Normal	Weak	Yes

```
x_train = data_train[['outlook','temp','humidity','wind']]
print(x_train)
y_train = data_train['play']
print(y_train)
```

```
      outlook  temp  humidity  wind
0     Sunny   Hot     High   Weak
1     Sunny   Hot     High  Strong
2  Overcast   Hot     High   Weak
3       Rain  Mild     High   Weak
4       Rain  Cool   Normal   Weak
5       Rain  Cool   Normal  Strong
6  Overcast  Cool   Normal  Strong
7     Sunny  Mild     High   Weak
8     Sunny  Cool   Normal   Weak
9       Rain  Mild   Normal   Weak
0       No
1       No
2     Yes
3     Yes
4     Yes
5       No
6     Yes
7       No
8     Yes
9     Yes
Name: play, dtype: object
```

```
data_test = pd.read_csv("/content/play_tennis_test.csv")
data_test.head()
```

	day	outlook	temp	humidity	wind	play
0	D11	Sunny	Mild	Normal	Strong	Yes
1	D12	Overcast	Mild	High	Strong	Yes
2	D13	Overcast	Hot	Normal	Weak	Yes
3	D14	Rain	Mild	High	Strong	No

```
x_test = data_test[['outlook','temp','humidity','wind']]
print(x_test)
y_test = data_test['play']
```

```
      outlook  temp humidity   wind
0      Sunny  Mild   Normal  Strong
1  Overcast  Mild    High  Strong
2  Overcast  Hot    Normal   Weak
3      Rain  Mild    High  Strong
```

```
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import recall_score
from sklearn.metrics import f1_score
```

```
le = LabelEncoder()
le.fit(y_train)
y_train_l=le.transform(y_train)
print(y_train_l)
le.fit(y_test)
y_test_l = le.transform(y_test)
print(y_test_l)
```

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

```
le.fit(x_train['outlook'])
print(x_train['outlook'])
```

```
x0_l=le.transform(x_train['outlook'])
x0_l1=le.transform(x_test['outlook'])
print(x0_l)
le.fit(x_train['temp'])
print(list(le.classes_))
x1_l=le.transform(x_train['temp'])
x1_l1=le.transform(x_test['temp'])
le.fit(x_train['humidity'])
x2_l=le.transform(x_train['humidity'])
x2_l1=le.transform(x_test['humidity'])
```

```

le.fit(x_train['wind'])
x3_l=le.transform(x_train['wind'])
x3_l1=le.transform(x_test['wind'])
x_train_l = np.array([x0_l,x1_l,x2_l,x3_l])
x_test_l = np.array([x0_l1,x1_l1,x2_l1,x3_l1])
x_test_l = x_test_l.transpose()
print("X test data:",x_test_l)
x_train_l = x_train_l.transpose()
print("X train data",x_train_l)

```

```

0      Sunny
1      Sunny
2      Overcast
3      Rain
4      Rain
5      Rain
6      Overcast
7      Sunny
8      Sunny
9      Rain
Name: outlook, dtype: object
[2 2 0 1 1 1 0 2 2 1]
['Cool', 'Hot', 'Mild']
X test data: [[2 2 1 0]
 [0 2 0 0]
 [0 1 1 1]
 [1 2 0 0]]
X train data [[2 1 0 1]
 [2 1 0 0]
 [0 1 0 1]
 [1 2 0 1]
 [1 0 1 1]
 [1 0 1 0]
 [0 0 1 0]
 [2 2 0 1]
 [2 0 1 1]
 [1 2 1 1]]

```

```
gnd = GaussianNB()
```

```
gnd.fit(x_train_l,y_train_l)
```

```
GaussianNB()
```

```

y_pred=gnd.predict(x_test_l)
print(y_pred)
print(y_test_l)
accuracy_score(y_test_l,y_pred)*100

```

```
[0 1 1 0]  
[1 1 1 0]  
75.0
```

```
confusion_matrix(y_test_1,y_pred)
```

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

```
recall_score(y_test_1,y_pred)
```

```
0.6666666666666666
```

```
f1_score(y_test_1, y_pred, average='macro')
```

```
0.7333333333333334
```

[Colab paid products](#) - [Cancel contracts here](#)

