

```
X=[
    ['Sunny','Hot','High','Weak'],
    ['Sunny','Hot','High','Strong'],
    ['Overcast','Hot','High','Weak'],
    ['Rain','Mild','High','Weak'],
    ['Rain','Cool','Normal','Weak'],
    ['Rain','Cool','Normal','Strong'],
    ['Overcast','Cool','Normal','Strong'],
    ['Sunny','Mild','High','Weak'],
    ['Sunny','Cool','Normal','Weak'],
    ['Rain','Mild','Normal','Weak'],
    ['Sunny','Mild','Normal','Strong'],
    ['Overcast','Mild','High','Strong'],
    ['Overcast','Hot','Normal','Weak'],
    ['Rain','Mild','High','Strong']
]
```

```
y=['No','No','Yes','Yes','Yes','No','Yes','No','Yes','Yes','Yes','Yes','Yes','No']
```

```
C=list(set(y))
```

```
def P(x,c):
```

```
    p=y.count(c)/len(y)
```

```
    for i in range(len(x)):
```

```
        p*=(sum(1 for j in range(len(X)) if X[j][i]==x[i] and y[j]==c)+1)/(y.count(c)+len(set(r[i] for r in X)))
```

```
    return p
```

```
pred=[max(C,key=lambda c:P(x,c)) for x in X]
```

```
cm=[[0,0],[0,0]]
```

```
for a,p in zip(y,pred):
```

```
    cm[C.index(a)][C.index(p)]+=1
```

```
print("Confusion Matrix:")  
  
for r in cm: print(r)  
  
print("Accuracy:",sum(cm[i][i] for i in range(2))/len(y))
```

OUTPUT:

```
>>> ===== RESTART: C:/Users/prast/OneDrive/Desktop/ML LAB/EXP 6.py =====  
Confusion Matrix:  
[4, 1]  
[0, 9]  
Accuracy: 0.9285714285714286  
>>>
```