

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
```

```
In [2]: df=pd.read_csv(r"C:\Users\mouni\Downloads\drug200.csv")
df
```

Out[2]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY
...
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

200 rows × 6 columns

```
In [3]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 6 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Age             200 non-null   int64
1   Sex             200 non-null   object
2   BP              200 non-null   object
3   Cholesterol      200 non-null   object
4   Na_to_K         200 non-null   float64
5   Drug            200 non-null   object
dtypes: float64(1), int64(1), object(4)
memory usage: 9.5+ KB
```

```
In [4]: df['Age'].value_counts()
```

```
Out[4]: Age
47      8
23      7
28      7
49      7
39      6
32      6
50      5
37      5
58      5
60      5
22      5
34      4
72      4
51      4
42      4
26      4
24      4
74      4
67      4
68      4
61      4
56      4
20      4
36      4
45      4
41      4
31      4
43      4
65      4
57      4
53      3
40      3
70      3
59      3
16      3
38      3
15      3
69      3
35      3
18      3
64      3
52      2
55      2
62      2
19      2
29      2
66      2
73      2
46      2
48      2
54      1
17      1
33      1
63      1
30      1
21      1
```

```
25      1
Name: count, dtype: int64
```

```
In [5]: df['BP'].value_counts()
```

```
Out[5]: BP
HIGH      77
LOW       64
NORMAL    59
Name: count, dtype: int64
```

```
In [6]: convert={"Cholesterol":{"HIGH":1,"NORMAL":0}}
df=df.replace(convert)
df
```

```
Out[6]:
```

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	1	25.355	drugY
1	47	M	LOW	1	13.093	drugC
2	47	M	LOW	1	10.114	drugC
3	28	F	NORMAL	1	7.798	drugX
4	61	F	LOW	1	18.043	drugY
...
195	56	F	LOW	1	11.567	drugC
196	16	M	LOW	1	12.006	drugC
197	52	M	NORMAL	1	9.894	drugX
198	23	M	NORMAL	0	14.020	drugX
199	40	F	LOW	0	11.349	drugX

200 rows × 6 columns

```
In [7]: convert={"BP":{"HIGH":1,"LOW":2,"NORMAL":3}}
df=df.replace(convert)
df
```

Out[7]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	1	1	25.355	drugY
1	47	M	2	1	13.093	drugC
2	47	M	2	1	10.114	drugC
3	28	F	3	1	7.798	drugX
4	61	F	2	1	18.043	drugY
...
195	56	F	2	1	11.567	drugC
196	16	M	2	1	12.006	drugC
197	52	M	3	1	9.894	drugX
198	23	M	3	0	14.020	drugX
199	40	F	2	0	11.349	drugX

200 rows × 6 columns

```
In [12]: x=["Age","BP","Cholesterol","Na_to_K"]
Y=["Yes","No"]
all_inputs=df[x]
all_classes=df["Drug"]
```

```
In [13]: (x_train,x_test,y_train,y_test)=train_test_split(all_inputs,all_classes,train_size=
```

```
In [14]: clf=DecisionTreeClassifier(random_state=0)
```

```
In [15]: clf.fit(x_train,y_train)
```

Out[15]:

```
DecisionTreeClassifier
DecisionTreeClassifier(random_state=0)
```

```
In [16]: score=clf.score(x_test,y_test)
print(score)
```

0.26262626262626265

In []:

