

NBA (wine dataset)

December 18, 2022

```
[2]: import numpy as np
import pandas as pd
```

```
[3]: from sklearn import datasets
wine = datasets.load_wine()
```

```
[4]: print ("Features: ", wine.feature_names)
```

```
Features:  ['alcohol', 'malic_acid', 'ash', 'alcalinity_of_ash', 'magnesium',
'total_phenols', 'flavanoids', 'nonflavanoid_phenols', 'proanthocyanins',
'color_intensity', 'hue', 'od280/od315_of_diluted_wines', 'proline']
```

```
[5]: print ("Labels: ", wine.target_names)
```

```
Labels:  ['class_0' 'class_1' 'class_2']
```

```
[8]: X=pd.DataFrame(wine['data'])

print(X.head())
```

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | \ |
|---|-------|------|------|------|-------|------|------|------|------|------|------|------|---|
| 0 | 14.23 | 1.71 | 2.43 | 15.6 | 127.0 | 2.80 | 3.06 | 0.28 | 2.29 | 5.64 | 1.04 | 3.92 | |
| 1 | 13.20 | 1.78 | 2.14 | 11.2 | 100.0 | 2.65 | 2.76 | 0.26 | 1.28 | 4.38 | 1.05 | 3.40 | |
| 2 | 13.16 | 2.36 | 2.67 | 18.6 | 101.0 | 2.80 | 3.24 | 0.30 | 2.81 | 5.68 | 1.03 | 3.17 | |
| 3 | 14.37 | 1.95 | 2.50 | 16.8 | 113.0 | 3.85 | 3.49 | 0.24 | 2.18 | 7.80 | 0.86 | 3.45 | |
| 4 | 13.24 | 2.59 | 2.87 | 21.0 | 118.0 | 2.80 | 2.69 | 0.39 | 1.82 | 4.32 | 1.04 | 2.93 | |

| | 12 |
|---|--------|
| 0 | 1065.0 |
| 1 | 1050.0 |
| 2 | 1185.0 |
| 3 | 1480.0 |
| 4 | 735.0 |

```
[7]: print(wine.data.shape)
```

```
(178, 13)
```

```
[9]: y=print (wine.target)
```

```
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2]
```

```
[10]: from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(wine.data, wine.target,
↳test_size=0.30,random_state=109)
```

```
[11]: from sklearn.naive_bayes import GaussianNB
gnb = GaussianNB()
gnb.fit(X_train, y_train)
y_pred = gnb.predict(X_test)

print(y_pred)
```

```
[0 0 1 2 0 1 0 0 1 0 2 2 2 2 0 1 1 0 0 1 2 1 0 2 0 0 1 2 0 1 2 1 1 0 1 1 0
 2 2 0 2 1 0 0 0 2 2 0 1 1 2 0 0 2]
```

```
[12]: from sklearn import metrics
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
```

Accuracy: 0.9074074074074074

```
[13]: from sklearn.metrics import confusion_matrix

cm=np.array(confusion_matrix(y_test,y_pred))

cm
```

```
[13]: array([[20,  1,  0],
           [ 2, 15,  2],
           [ 0,  0, 14]], dtype=int64)
```

```
[ ]:
```