```python
In [6]:  import pandas as pd
         import numpy as np
```

```python
In [7]:  from sklearn.datasets import load_wine
```

```python
In [8]:  winedata = load_wine()
```

```python
In [9]:  winedata.keys()
```

```
Out[9]:  dict_keys(['data', 'target', 'frame', 'target_names', 'DESCR', 'feature_names'])
```

```python
In [10]:  dfwine = pd.DataFrame(winedata["data"], columns =winedata ["feature_names"])
```

```python
In [11]:  dfwine
```

Out[11]:

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity	hue	od280/od315_of_diluted_wines	proline
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	1065.0
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	1050.0
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	1185.0
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	1480.0
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	735.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
173	13.71	5.65	2.45	20.5	95.0	1.68	0.61	0.52	1.06	7.70	0.64	1.74	740.0
174	13.40	3.91	2.48	23.0	102.0	1.80	0.75	0.43	1.41	7.30	0.70	1.56	750.0
175	13.27	4.28	2.26	20.0	120.0	1.59	0.69	0.43	1.35	10.20	0.59	1.56	835.0
176	13.17	2.59	2.37	20.0	120.0	1.65	0.68	0.53	1.46	9.30	0.60	1.62	840.0
177	14.13	4.10	2.74	24.5	96.0	2.05	0.76	0.56	1.35	9.20	0.61	1.60	560.0

178 rows × 13 columns

```python
In [12]:  X = dfwine.copy()
```

```python
In [13]:  X.isnull().sum()
```

```
Out[13]:  alcohol                         0
          malic_acid                      0
          ash                             0
          alcalinity_of_ash               0
          magnesium                       0
          total_phenols                   0
          flavanoids                      0
          nonflavanoid_phenols            0
          proanthocyanins                 0
          color_intensity                 0
          hue                             0
          od280/od315_of_diluted_wines    0
          proline                         0
          dtype: int64
```

```python
In [14]:  X["alcohol"].skew()
```

```
Out[14]:  -0.051482331077134784
```

```python
In [15]:  y = winedata["target"]
```

```python
In [16]:  y
```

```
Out[16]:  array([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, 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, 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])
```

```python
In [17]:  from sklearn.model_selection import train_test_split
```

```python
In [18]:  Xtrain, Xtest, ytrain, ytest = train_test_split(X,y, test_size=0.2)
```

```python
In [19]:  Xtrain.shape, Xtest.shape, X.shape
```

```
Out[19]:  ((142, 13), (36, 13), (178, 13))
```

```python
In [22]:  from sklearn.tree import DecisionTreeClassifier
```

```python
In [25]:  dtent = DecisionTreeClassifier(criterion="entropy")
```

```python
In [26]:  dtgini = DecisionTreeClassifier()
```

```python
In [27]:  dtent.fit(Xtrain,ytrain)
```

```
Out[27]:  DecisionTreeClassifier(criterion='entropy')
```

```python
In [28]:  dtgini.fit(Xtrain, ytrain)
```

```
Out[28]:  DecisionTreeClassifier()
```

```python
In [29]:  dtent.score(Xtest,ytest)
```

```
Out[29]:  0.9166666666666666
```

```python
In [30]:  dtgini.score(Xtest,ytest)
```

```
Out[30]:  0.9444444444444444
```

```python
In [31]:  from sklearn.model_selection import KFold, cross_val_score
```

```python
In [32]:  kf = KFold(n_splits=10)
```

```python
In [33]:  giniscore = cross_val_score(DecisionTreeClassifier(), X,y, cv= kf)
```

```python
In [34]:  giniscore
```

```
Out[34]:  array([1.        , 0.83333333, 0.88888889, 0.61111111, 0.77777778,
                 0.94444444, 0.94444444, 0.72222222, 0.88235294, 0.94117647])
```

```python
In [35]:  giniscore.mean()
```

```
Out[35]:  0.8545751633986928
```

```python
In [36]:  entroscore = cross_val_score(DecisionTreeClassifier(criterion="entropy"),X,y, cv = kf)
```

```python
In [37]:  entroscore.mean()
```

```
Out[37]:  0.8872549019607842
```

```python
In [ ]:
```