In [16]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import warnings
warnings.filterwarnings("ignore")
```

In [12]:

```
df = pd.read_csv("/Users/sudhanshubiswal/Project/100-days-of-machine-learning-ma
                 header=None,usecols=[0,1,2])
df.columns=['Class label', 'Alcohol', 'Malic acid']

```

In [9]:

```
df
```

Out[9]:

	Class label	Alcohol	Malic acid
1	7.4	0.7	0.0
2	7.8	0.88	0.0
3	7.8	0.76	0.04
4	11.2	0.28	0.56
5	7.4	0.7	0.0
...	...	...	...
1139	6.3	0.51	0.13
1140	6.8	0.62	0.08
1141	6.2	0.6	0.08
1142	5.9	0.55	0.1
1143	5.9	0.645	0.12

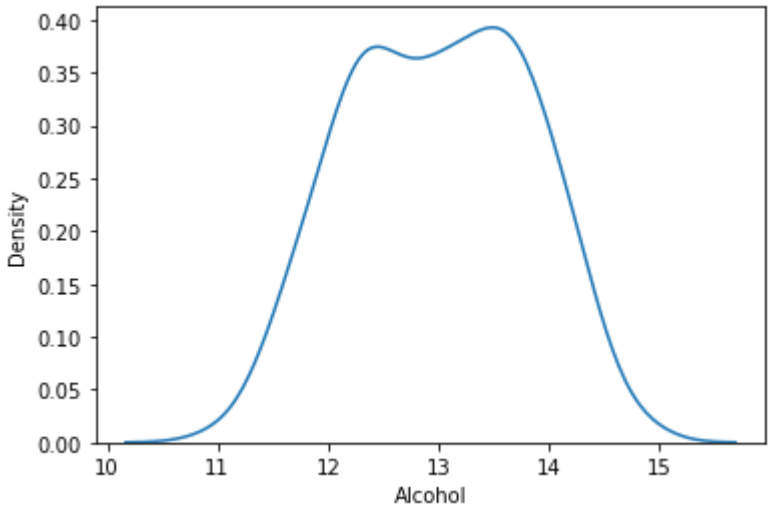
1143 rows × 3 columns

In [13]:

```
1 sns.kdeplot(df['Alcohol'])
```

Out[13]:

```
<AxesSubplot:xlabel='Alcohol', ylabel='Density'>
```

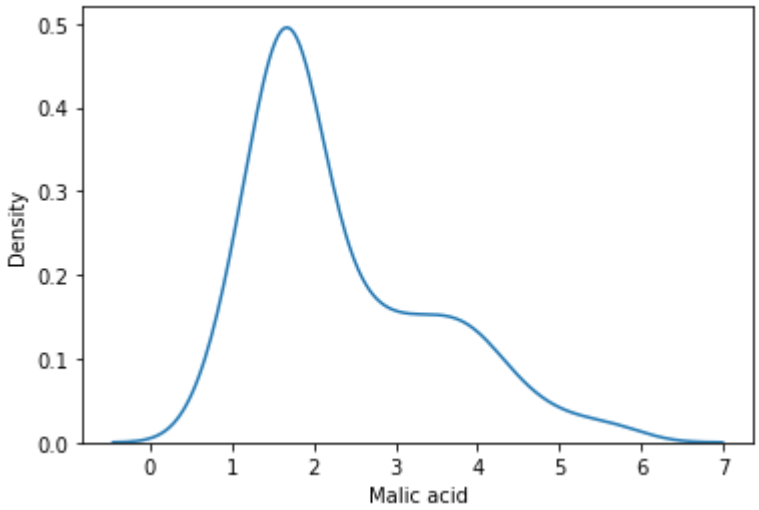


In [14]:

```
1 sns.kdeplot(df["Malic acid"])
```

Out[14]:

```
<AxesSubplot:xlabel='Malic acid', ylabel='Density'>
```

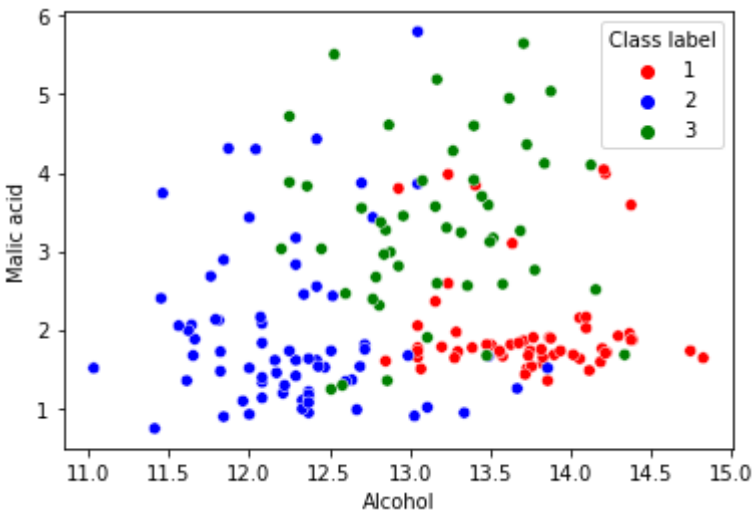


In [17]:

```
1 color_dict={1:'red',3:'green',2:'blue'}
2 sns.scatterplot(df['Alcohol'],df['Malic acid'],hue=df['Class label'],palette=col
```

Out[17]:

```
<AxesSubplot:xlabel='Alcohol', ylabel='Malic acid'>
```



In [18]:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(df.drop('Class label', axis=
                                                    df['Class label'],
                                                    test_size=0.3,
                                                    random_state=0)

X_train.shape, X_test.shape
```

Out[18]:

```
((124, 2), (54, 2))
```

In [19]:

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

# fit the scaler to the train set, it will learn the parameters
scaler.fit(X_train)

# transform train and test sets
X_train_scaled = scaler.transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

In [20]:

```
X_train_scaled = pd.DataFrame(X_train_scaled, columns=X_train.columns)
X_test_scaled = pd.DataFrame(X_test_scaled, columns=X_test.columns)
```

In [21]:

```
np.round(X_train.describe(), 1)
```

Out[21]:

	Alcohol	Malic acid
count	124.0	124.0
mean	13.0	2.4
std	0.8	1.1
min	11.0	0.9
25%	12.4	1.6
50%	13.0	1.9
75%	13.6	3.2
max	14.8	5.6

In [22]:

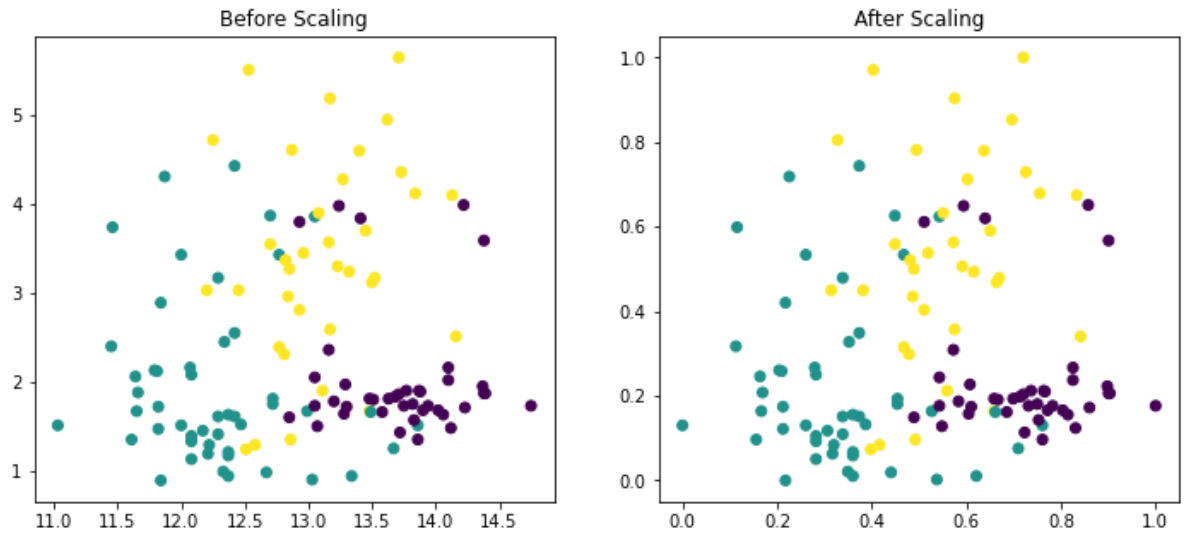
```
1 np.round(X_train_scaled.describe(), 1)
```

Out[22]:

	Alcohol	Malic acid
count	124.0	124.0
mean	0.5	0.3
std	0.2	0.2
min	0.0	0.0
25%	0.4	0.2
50%	0.5	0.2
75%	0.7	0.5
max	1.0	1.0

In [23]:

```
1 fig, (ax1, ax2) = plt.subplots(ncols=2, figsize=(12, 5))
2
3 ax1.scatter(X_train['Alcohol'], X_train['Malic acid'],c=y_train)
4 ax1.set_title("Before Scaling")
5 ax2.scatter(X_train_scaled['Alcohol'], X_train_scaled['Malic acid'],c=y_train)
6 ax2.set_title("After Scaling")
7 plt.show()
```

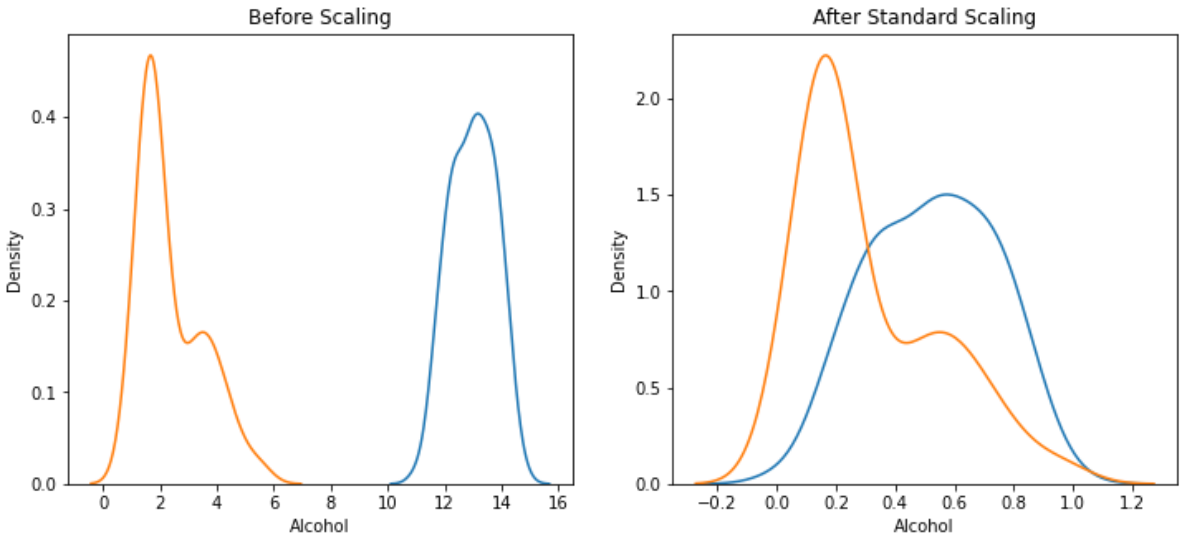


In [24]:

```
fig, (ax1, ax2) = plt.subplots(ncols=2, figsize=(12, 5))

# before scaling
ax1.set_title('Before Scaling')
sns.kdeplot(X_train['Alcohol'], ax=ax1)
sns.kdeplot(X_train['Malic acid'], ax=ax1)

# after scaling
ax2.set_title('After Standard Scaling')
sns.kdeplot(X_train_scaled['Alcohol'], ax=ax2)
sns.kdeplot(X_train_scaled['Malic acid'], ax=ax2)
plt.show()
```

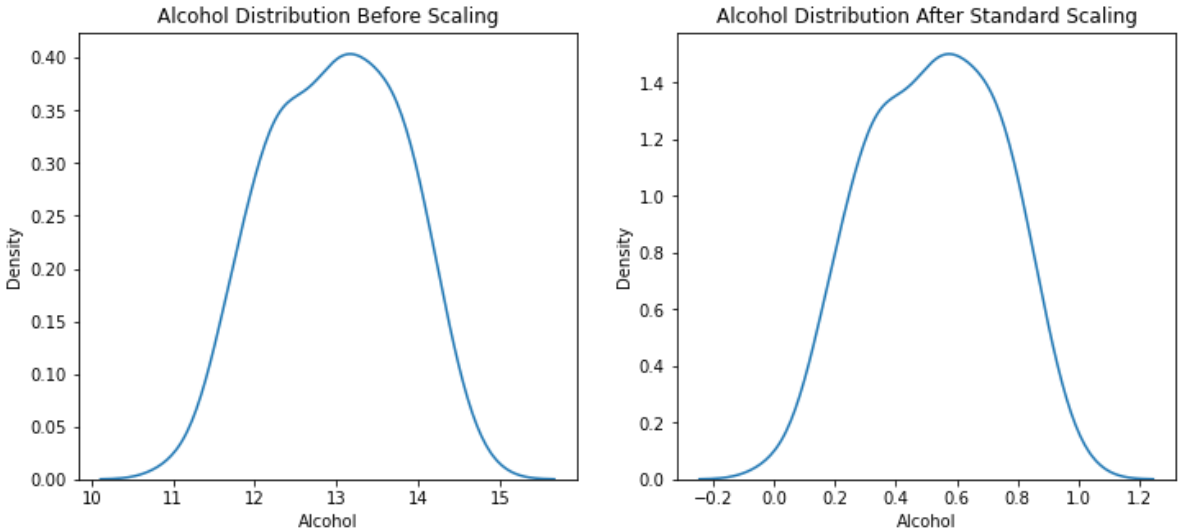


In [25]:

```
fig, (ax1, ax2) = plt.subplots(ncols=2, figsize=(12, 5))

# before scaling
ax1.set_title('Alcohol Distribution Before Scaling')
sns.kdeplot(X_train['Alcohol'], ax=ax1)

# after scaling
ax2.set_title('Alcohol Distribution After Standard Scaling')
sns.kdeplot(X_train_scaled['Alcohol'], ax=ax2)
plt.show()
```



In [26]:

```
fig, (ax1, ax2) = plt.subplots(ncols=2, figsize=(12, 5))

# before scaling
ax1.set_title('Malic acid Distribution Before Scaling')
sns.kdeplot(X_train['Malic acid'], ax=ax1)

# after scaling
ax2.set_title('Malic acid Distribution After Standard Scaling')
sns.kdeplot(X_train_scaled['Malic acid'], ax=ax2)
plt.show()
```