



```
In [1]: import pandas as pd
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt
```

```
In [2]: df = pd.read_csv('Wine.csv')
```

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

```
Out[3]: Index(['Alcohol', 'Malic_Acid', 'Ash', 'Ash_Alcanyt', 'Magnesium',
       'Total_Phenols', 'Flavanoids', 'Nonflavanoid_Phenols',
       'Proanthocyanins', 'Color_Intensity', 'Hue', 'OD280', 'Proline',
       'Customer_Segment'],
      dtype='object')
```

```
In [7]: df.head(5)
```

```
Out[7]:   Alcohol  Malic_Acid  Ash  Ash_Alcanyt  Magnesium  Total_Phenols  Flavanoid
0    14.23        1.71  2.43         15.6      127          2.80        3.0
1    13.20        1.78  2.14         11.2      100          2.65        2.7
2    13.16        2.36  2.67         18.6      101          2.80        3.2
3    14.37        1.95  2.50         16.8      113          3.85        3.4
4    13.24        2.59  2.87         21.0      118          2.80        2.6
```

```
In [8]: df.Customer_Segment.unique()
```

```
Out[8]: array([1, 2, 3])
```

```
In [9]: print(df.isnull().sum())
```

Alcohol	0
Malic_Acid	0
Ash	0
Ash_Alcanyt	0
Magnesium	0
Total_Phenols	0
Flavanoids	0
Nonflavanoid_Phenols	0
Proanthocyanins	0
Color_Intensity	0
Hue	0
OD280	0
Proline	0
Customer_Segment	0

```
dtype: int64
```

```
In [10]: x = df.drop('Customer_Segment', axis=1)
y = df['Customer_Segment']
```

```
In [12]: for col in x.columns:  
    sc = StandardScaler()  
    x[col] = sc.fit_transform(x[[col]])
```

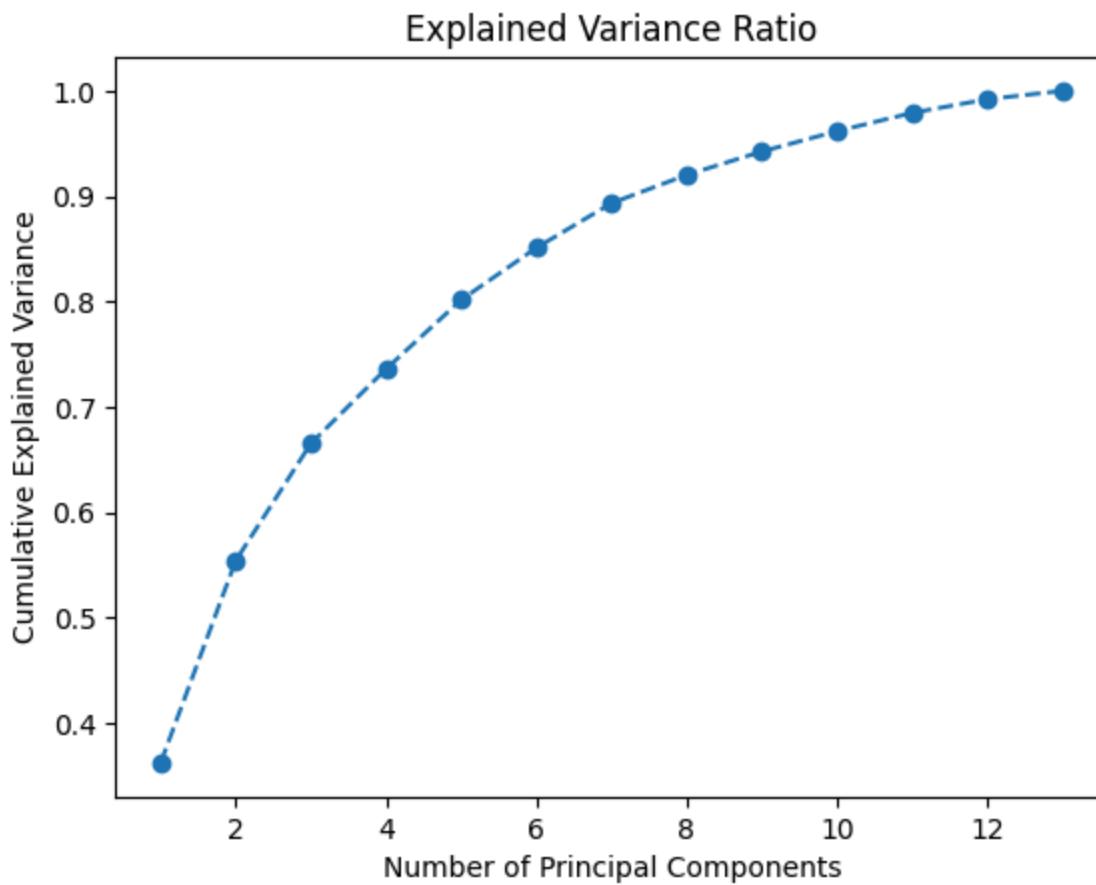
```
In [13]: x.head(5)
```

```
Out[13]:   Alcohol  Malic_Acid      Ash  Ash_Alcanity  Magnesium  Total_Phenols  Flav  
0  1.518613  -0.562250  0.232053  -1.169593  1.913905  0.808997  1  
1  0.246290  -0.499413  -0.827996  -2.490847  0.018145  0.568648  0  
2  0.196879  0.021231  1.109334  -0.268738  0.088358  0.808997  1  
3  1.691550  -0.346811  0.487926  -0.809251  0.930918  2.491446  1  
4  0.295700  0.227694  1.840403  0.451946  1.281985  0.808997  0
```

```
In [14]: pca = PCA()  
x_pca = pca.fit_transform(x)
```

```
In [15]: explained_variance_ratio = pca.explained_variance_ratio_
```

```
In [18]: plt.plot(range(1, len(explained_variance_ratio) + 1), explained_variance_ratio)  
plt.xlabel('Number of Principal Components')  
plt.ylabel('Cumulative Explained Variance')  
plt.title('Explained Variance Ratio')  
plt.show()
```



```
In [18]: n_components = 12
pca = PCA(n_components = n_components)
x_pca = pca.fit_transform(x)
```

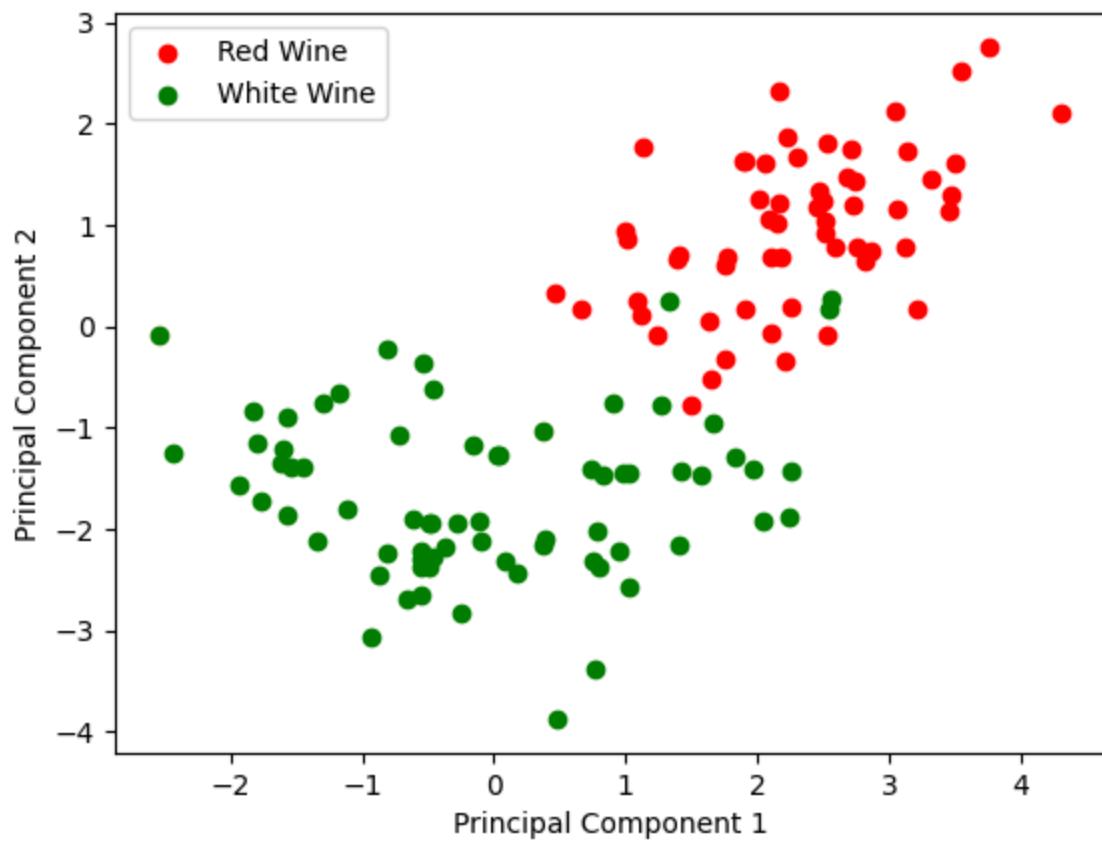
```
In [19]: x_pca.shape
```

```
Out[19]: (178, 12)
```

```
In [21]: red_indices = y[y == 1].index
white_indices = y[y == 2].index

plt.scatter(x_pca[red_indices, 0], x_pca[red_indices, 1], c='red', label='Red'
plt.scatter(x_pca[white_indices, 0], x_pca[white_indices, 1], c='green', label='White')
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
plt.legend()
plt.title('PCA: Red Wine vs White Wine')
plt.show()
```

PCA: Red Wine vs White Wine



In [ ]: