

```

import pandas as pd
import numpy as np
from sklearn.datasets import load_wine
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA

# Load dataset
wine = load_wine()
df = pd.DataFrame(wine.data, columns=wine.feature_names)
df['target'] = wine.target

```

```
df.head()
```

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium
0	14.23	1.71	2.43	15.6	127.0
1	13.20	1.78	2.14	11.2	100.0
2	13.16	2.36	2.67	18.6	101.0
3	14.37	1.95	2.50	16.8	113.0
4	13.24	2.59	2.87	21.0	118.0

	flavanoids	nonflavanoid_phenols	proanthocyanins	color_intensity
0	3.06	0.28	2.29	5.64
1	2.76	0.26	1.28	4.38
2	3.24	0.30	2.81	5.68
3	3.49	0.24	2.18	7.80
4	2.69	0.39	1.82	4.32

	od280/od315_of_diluted_wines	proline	target
0	3.92	1065.0	0
1	3.40	1050.0	0
2	3.17	1185.0	0
3	3.45	1480.0	0
4	2.93	735.0	0

```
# Display the basic statistics using describe()
df.describe()
```

	alcohol	malic_acid	ash	alcalinity_of_ash
magnesium \				
count	178.000000	178.000000	178.000000	178.000000
mean	13.000618	2.336348	2.366517	19.494944
std	0.811827	1.117146	0.274344	3.339564
min	11.030000	0.740000	1.360000	10.600000
25%	12.362500	1.602500	2.210000	17.200000
50%	13.050000	1.865000	2.360000	19.500000
75%	13.677500	3.082500	2.557500	21.500000
max	14.830000	5.800000	3.230000	30.000000

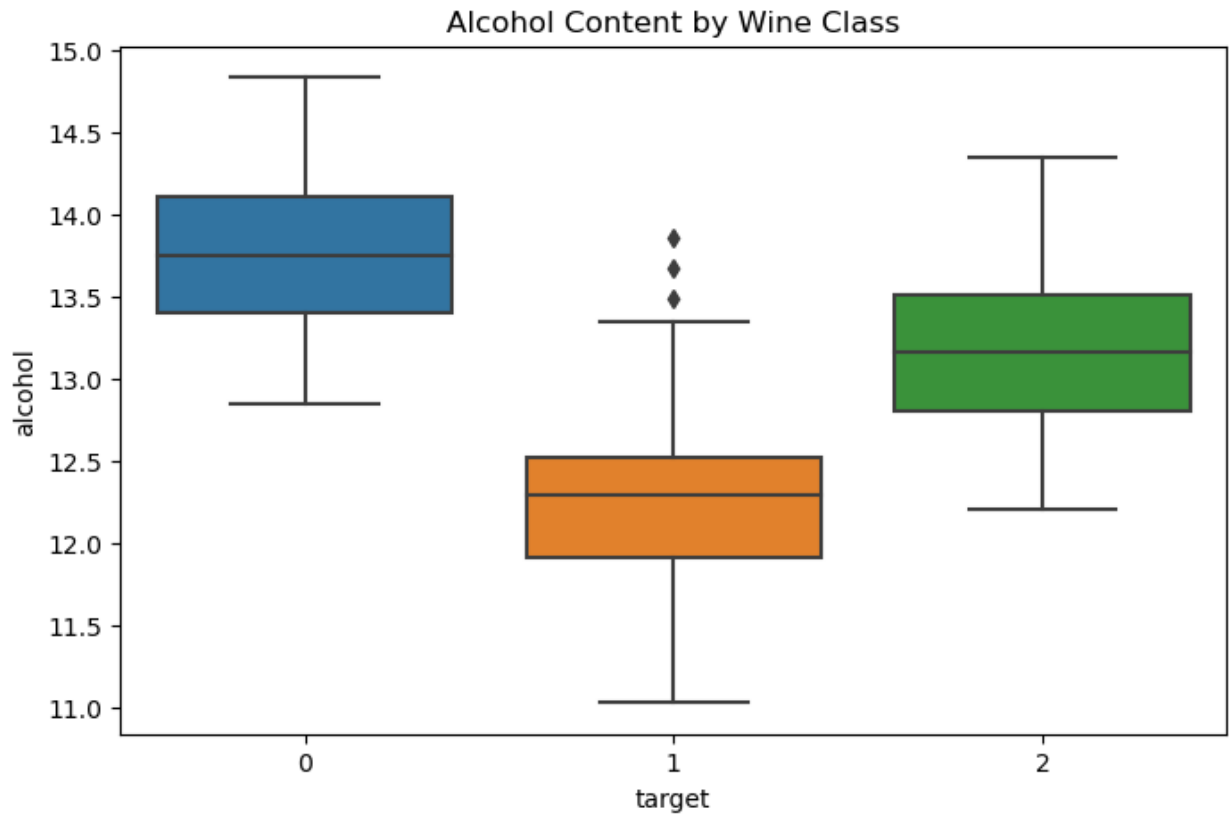
	total_phenols	flavanoids	nonflavanoid_phenols
proanthocyanins \			
count	178.000000	178.000000	178.000000
mean	2.295112	2.029270	0.361854
std	0.625851	0.998859	0.124453
min	0.980000	0.340000	0.130000
25%	1.742500	1.205000	0.270000
50%	2.355000	2.135000	0.340000
75%	2.800000	2.875000	0.437500
max	3.880000	5.080000	0.660000

	color_intensity	hue	od280/od315_of_diluted_wines
proline \			
count	178.000000	178.000000	178.000000
mean	5.058090	0.957449	2.611685
std	2.318286	0.228572	0.709990

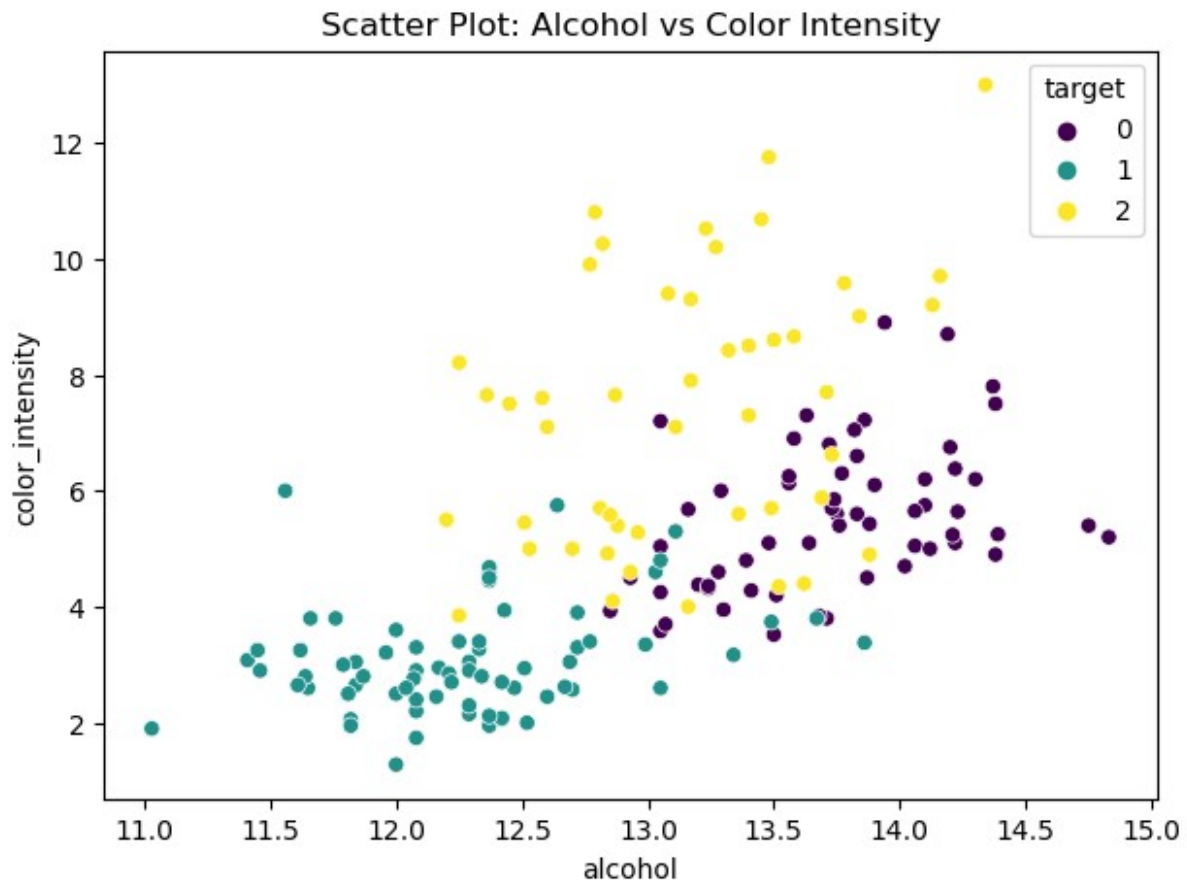
min	1.280000	0.480000	1.270000
278.000000			
25%	3.220000	0.782500	1.937500
500.500000			
50%	4.690000	0.965000	2.780000
673.500000			
75%	6.200000	1.120000	3.170000
985.000000			
max	13.000000	1.710000	4.000000
1680.000000			

	target
count	178.000000
mean	0.938202
std	0.775035
min	0.000000
25%	0.000000
50%	1.000000
75%	2.000000
max	2.000000

```
# Create a boxplot grouped by output labels/classes
plt.figure(figsize=(8,5))
sns.boxplot(x='target', y='alcohol', data=df)
plt.title("Alcohol Content by Wine Class")
plt.show()
```

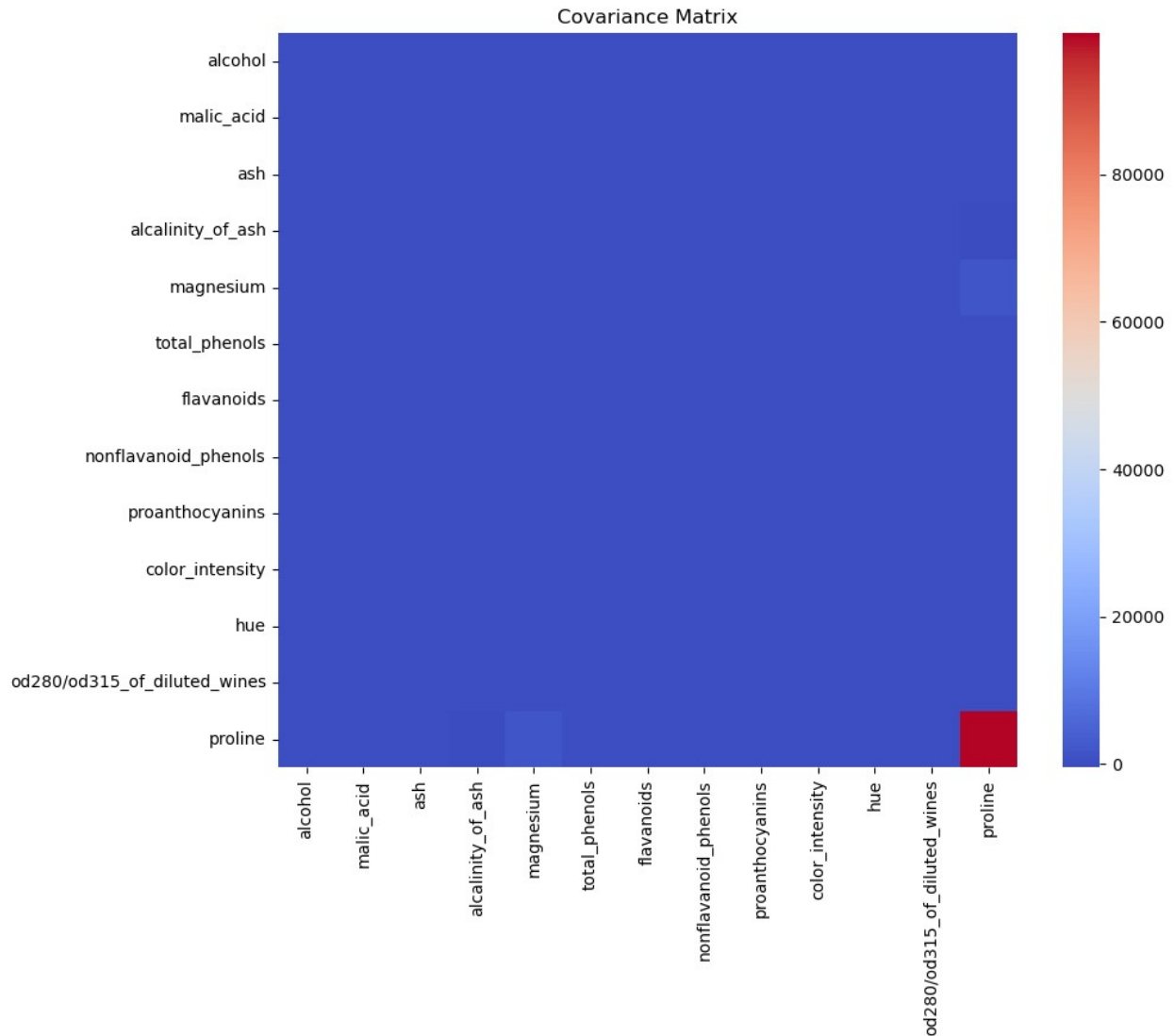


```
# Create a scatterplot using two variables
plt.figure(figsize=(7,5))
sns.scatterplot(x=df['alcohol'], y=df['color_intensity'],
hue=df['target'], palette='viridis')
plt.title("Scatter Plot: Alcohol vs Color Intensity")
plt.show()
```



```
# Plot the covariance matrix
cov_matrix = df.drop(columns=['target']).cov()

plt.figure(figsize=(10,8))
sns.heatmap(cov_matrix, cmap="coolwarm", annot=False)
plt.title("Covariance Matrix")
plt.show()
```



Apply data scaling

```
scaler = StandardScaler()
```

```
scaled = scaler.fit_transform(df.drop(columns=['target']))
```

```
scaled_df = pd.DataFrame(scaled, columns=wine.feature_names)
```

```
scaled_df.head()
```

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	\
0	1.518613	-0.562250	0.232053	-1.169593	1.913905	
1	0.246290	-0.499413	-0.827996	-2.490847	0.018145	
2	0.196879	0.021231	1.109334	-0.268738	0.088358	
3	1.691550	-0.346811	0.487926	-0.809251	0.930918	
4	0.295700	0.227694	1.840403	0.451946	1.281985	
	total_phenols	flavanoids	nonflavanoid_phenols	proanthocyanins	\	
0	0.808997	1.034819	-0.659563	1.224884		

1	0.568648	0.733629	-0.820719	-0.544721
2	0.808997	1.215533	-0.498407	2.135968
3	2.491446	1.466525	-0.981875	1.032155
4	0.808997	0.663351	0.226796	0.401404

	color_intensity	hue	od280/od315_of_diluted_wines	proline
0	0.251717	0.362177	1.847920	1.013009
1	-0.293321	0.406051	1.113449	0.965242
2	0.269020	0.318304	0.788587	1.395148
3	1.186068	-0.427544	1.184071	2.334574
4	-0.319276	0.362177	0.449601	-0.037874

Show better class separation using principal components (PCA)

```
pca = PCA(n_components=2)
```

```
pca_data = pca.fit_transform(scaled)
```

```
pca_df = pd.DataFrame(data=pca_data, columns=['PC1', 'PC2'])
```

```
pca_df['target'] = wine.target
```

```
plt.figure(figsize=(7,5))
```

```
sns.scatterplot(x='PC1', y='PC2', hue='target', data=pca_df,  
palette='rocket')
```

```
plt.title("PCA - Wine Class Separation")
```

```
plt.show()
```

