

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
data=pd.read_csv("C:/Users/SRMVEC/Downloads/dev wine dataset.csv")
```

```
data
```

	fixed acidity	volatile acidity	citric acid	residual sugar
0	7.4	0.70	0.00	1.9
0.076				
1	7.8	0.88	0.00	2.6
0.098				
2	7.8	0.76	0.04	2.3
0.092				
3	11.2	0.28	0.56	1.9
0.075				
4	7.4	0.70	0.00	1.9
0.076				
5	7.4	0.66	0.00	1.8
0.075				
6	7.9	0.60	0.06	1.6
0.069				
7	8.1	0.50	0.10	2.0
0.084				
8	6.9	0.78	0.05	2.5
0.070				
9	8.5	0.45	0.30	3.5
0.090				
10	7.0	0.65	0.10	1.4
0.079				
11	8.3	0.42	0.20	1.8
0.086				
12	9.0	0.30	0.40	4.0
0.058				
13	7.6	0.70	0.08	1.7
0.072				
14	7.3	0.60	0.15	2.1
0.075				
15	8.2	0.35	0.25	3.2
0.088				
16	8.0	0.35	0.25	3.2
0.888				
17	8.0	0.76	0.10	2.3
0.079				
18	7.2	0.60	0.02	2.1
0.095				
19	9.2	0.40	0.50	3.6

0.065				
20	6.8	0.90	0.00	1.5
0.071				
21	8.7	0.50	0.10	2.5
0.088				
22	7.1	0.68	0.04	1.9
0.077				
23	8.6	0.30	0.30	3.1
0.066				
24	7.5	0.72	0.20	2.4
0.074				
25	7.4	0.55	0.15	1.8
0.069				
26	9.1	0.42	0.40	4.2
0.055				
27	7.3	0.80	0.10	2.0
0.080				
28	8.4	0.49	0.25	3.0
0.085				
29	6.7	0.78	0.02	1.7
0.079				
30	8.9	0.36	0.30	2.8
0.067				
31	7.0	0.67	0.00	1.4
0.073				
32	9.3	0.40	0.35	4.5
0.061				

	free sulfur dioxide	total sulfur dioxide	density	pH
sulphates \				
0	11	34	0.9978	3.51
0.56				
1	25	67	0.9968	3.20
0.68				
2	15	54	0.9970	3.26
0.65				
3	17	60	0.9980	3.16
0.58				
4	11	34	0.9978	3.51
0.56				
5	13	40	0.9978	3.51
0.56				
6	15	59	0.9964	3.30
0.46				
7	19	52	0.9965	3.40
0.63				
8	22	58	0.9971	3.48
0.59				
9	10	45	0.9959	3.19

0.61				
10	18	50	0.9974	3.61
0.55				
11	24	63	0.9960	3.25
0.67				
12	16	38	0.9955	3.22
0.64				
13	14	46	0.9968	3.53
0.62				
14	12	42	0.9970	3.45
0.58				
15	20	55	0.9964	3.21
0.66				
16	20	55	0.9964	3.21
0.66				
17	14	55	0.9962	3.22
0.59				
18	20	60	0.9969	3.45
0.61				
19	18	50	0.9954	3.18
0.64				
20	22	64	0.9976	3.55
0.57				
21	15	45	0.9961	3.30
0.62				
22	10	42	0.9972	3.48
0.58				
23	19	48	0.9958	3.20
0.65				
24	11	49	0.9965	3.52
0.60				
25	14	53	0.9973	3.34
0.63				
26	16	41	0.9951	3.17
0.66				
27	17	62	0.9960	3.50
0.58				
28	12	47	0.9962	3.26
0.64				
29	23	59	0.9977	3.53
0.59				
30	15	44	0.9956	3.22
0.65				
31	13	50	0.9970	3.42
0.60				
32	18	55	0.9949	3.15
0.62				

alcohol quality Id

0	9.4	5	0
1	9.8	5	1
2	9.8	5	2
3	9.8	6	3
4	9.4	5	4
5	9.4	5	5
6	9.4	5	6
7	10.2	6	7
8	9.1	4	8
9	10.8	7	9
10	9.0	4	10
11	10.5	6	11
12	11.2	6	12
13	9.5	5	13
14	9.2	4	14
15	10.7	7	15
16	10.7	7	15
17	10.0	6	0
18	9.5	5	1
19	11.0	7	2
20	9.1	4	3
21	10.5	6	4
22	9.3	5	5
23	10.9	6	6
24	9.9	5	7
25	9.6	5	8
26	11.4	6	9
27	9.7	4	10
28	10.6	6	11
29	9.0	4	12
30	10.3	7	13
31	9.1	5	14
32	11.5	6	15

```
data.head()
```

	fixed acidity chlorides \	volatile acidity	citric acid	residual sugar
0	7.4	0.70	0.00	1.9
1	7.8	0.88	0.00	2.6
2	7.8	0.76	0.04	2.3
3	11.2	0.28	0.56	1.9
4	7.4	0.70	0.00	1.9

free sulfur dioxide	total sulfur dioxide	density	pH	sulphates
---------------------	----------------------	---------	----	-----------

\						
0	11	34	0.9978	3.51	0.56	
1	25	67	0.9968	3.20	0.68	
2	15	54	0.9970	3.26	0.65	
3	17	60	0.9980	3.16	0.58	
4	11	34	0.9978	3.51	0.56	

	alcohol	quality	Id
0	9.4	5	0
1	9.8	5	1
2	9.8	5	2
3	9.8	6	3
4	9.4	5	4

data.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 33 entries, 0 to 32

Data columns (total 13 columns):

#	Column	Non-Null Count	Dtype
---	-----	-----	-----
0	fixed acidity	33 non-null	float64
1	volatile acidity	33 non-null	float64
2	citric acid	33 non-null	float64
3	residual sugar	33 non-null	float64
4	chlorides	33 non-null	float64
5	free sulfur dioxide	33 non-null	int64
6	total sulfur dioxide	33 non-null	int64
7	density	33 non-null	float64
8	pH	33 non-null	float64
9	sulphates	33 non-null	float64
10	alcohol	33 non-null	float64
11	quality	33 non-null	int64
12	Id	33 non-null	int64

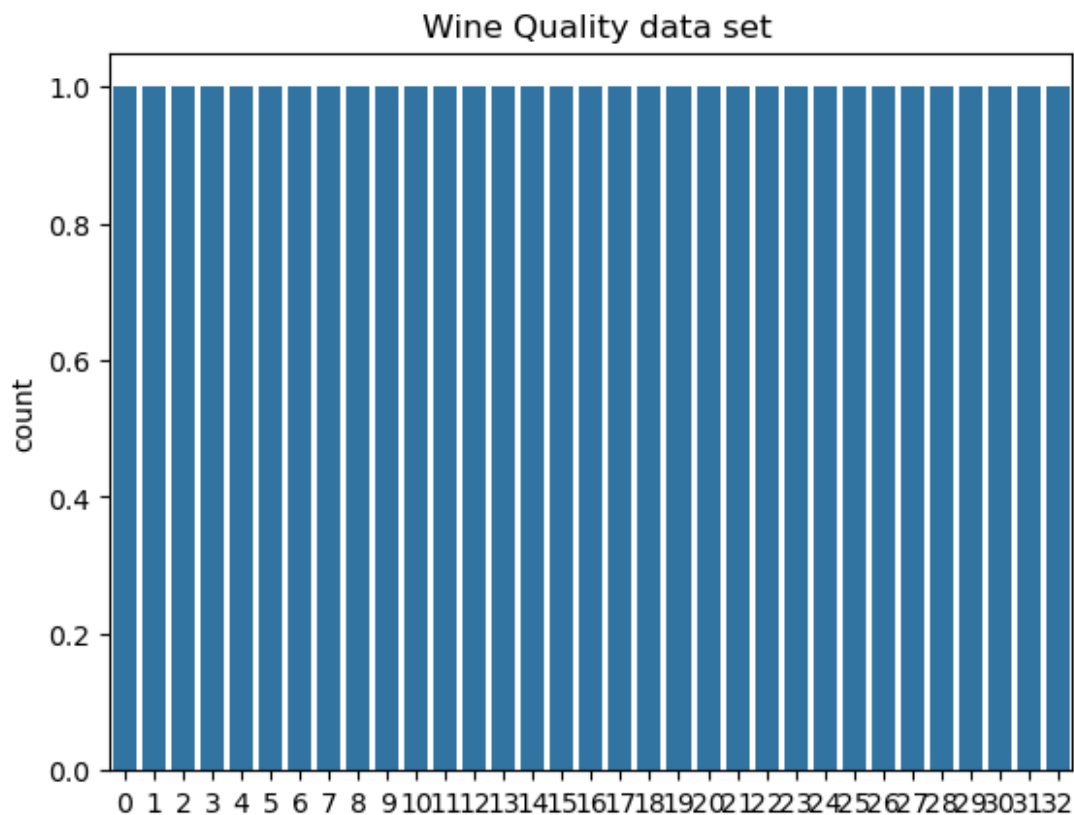
dtypes: float64(9), int64(4)

memory usage: 3.5 KB

data.describe()

	fixed acidity	volatile acidity	citric acid	residual sugar	\
count	33.000000	33.000000	33.000000	33.000000	
mean	7.969697	0.576061	0.162727	2.430303	
std	0.943227	0.179773	0.156250	0.833098	
min	6.700000	0.280000	0.000000	1.400000	
25%	7.300000	0.420000	0.040000	1.800000	
50%	7.800000	0.600000	0.100000	2.100000	

75%	8.500000	0.700000	0.250000	3.000000	
max	11.200000	0.900000	0.560000	4.500000	
	chlorides	free sulfur dioxide	total sulfur dioxide	density	
\					
count	33.000000	33.000000	33.000000	33.000000	
mean	0.101061	16.333333	50.787879	0.996609	
std	0.141639	4.165833	8.684604	0.000833	
min	0.055000	10.000000	34.000000	0.994900	
25%	0.070000	13.000000	45.000000	0.996000	
50%	0.076000	16.000000	50.000000	0.996500	
75%	0.085000	19.000000	58.000000	0.997200	
max	0.888000	25.000000	67.000000	0.998000	
	pH	sulphates	alcohol	quality	Id
count	33.000000	33.000000	33.000000	33.000000	33.000000
mean	3.348182	0.608788	9.978788	5.424242	7.727273
std	0.146937	0.044983	0.748648	0.969223	4.791090
min	3.150000	0.460000	9.000000	4.000000	0.000000
25%	3.210000	0.580000	9.400000	5.000000	4.000000
50%	3.300000	0.610000	9.800000	5.000000	8.000000
75%	3.500000	0.640000	10.600000	6.000000	12.000000
max	3.610000	0.680000	11.500000	7.000000	15.000000
sns.countplot (data ['quality'])					
plt.title (" Wine Quality data set")					
plt.show ()					

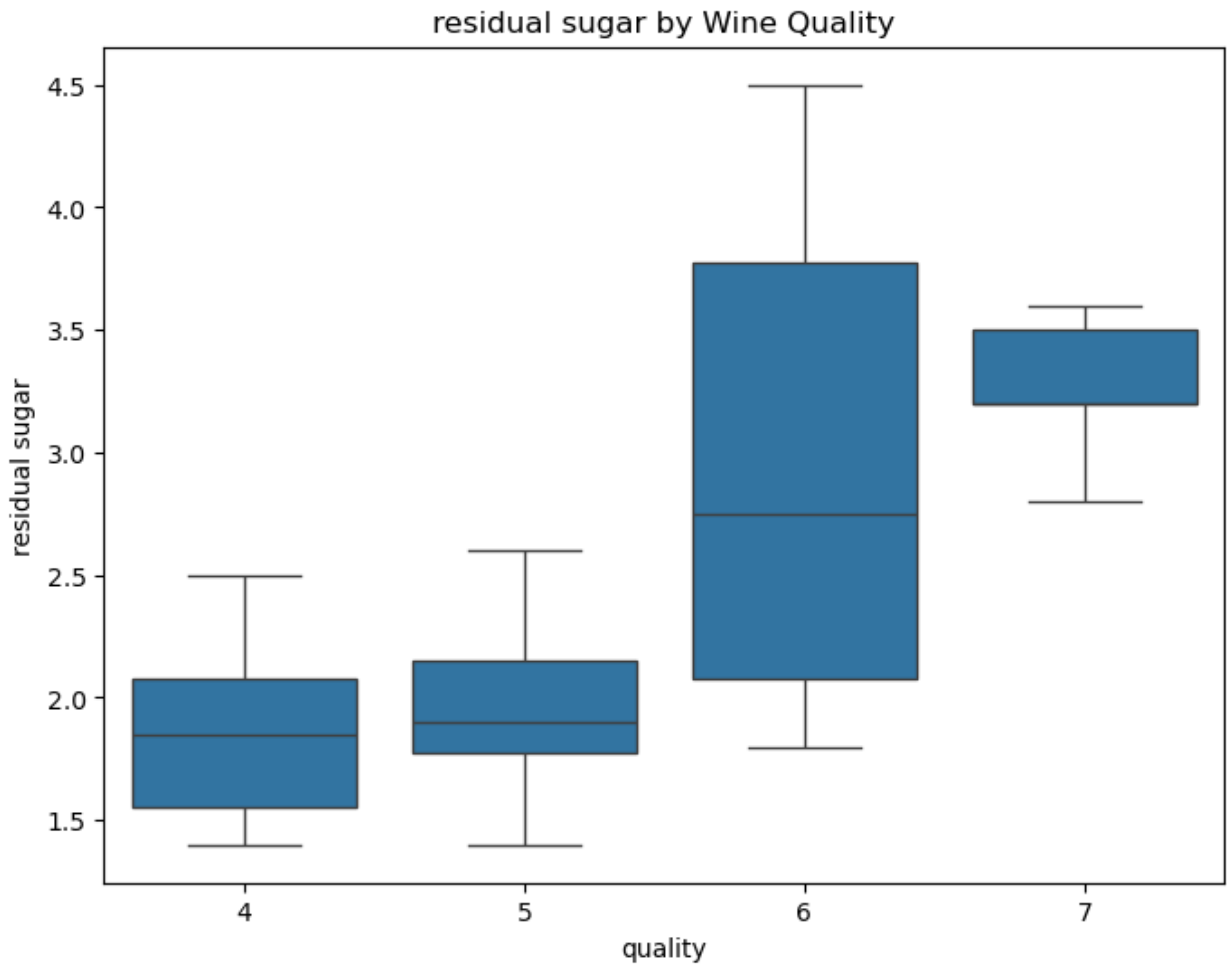


```
features = ['alcohol', 'volatile acidity', 'citric acid', 'residual
sugar']
for feature in features:
    plt.figure(figsize=(8, 6))
    sns.boxplot(x='quality', y=feature, data=data)
    plt.title(f'{feature} by Wine Quality')
    plt.show ( )
```

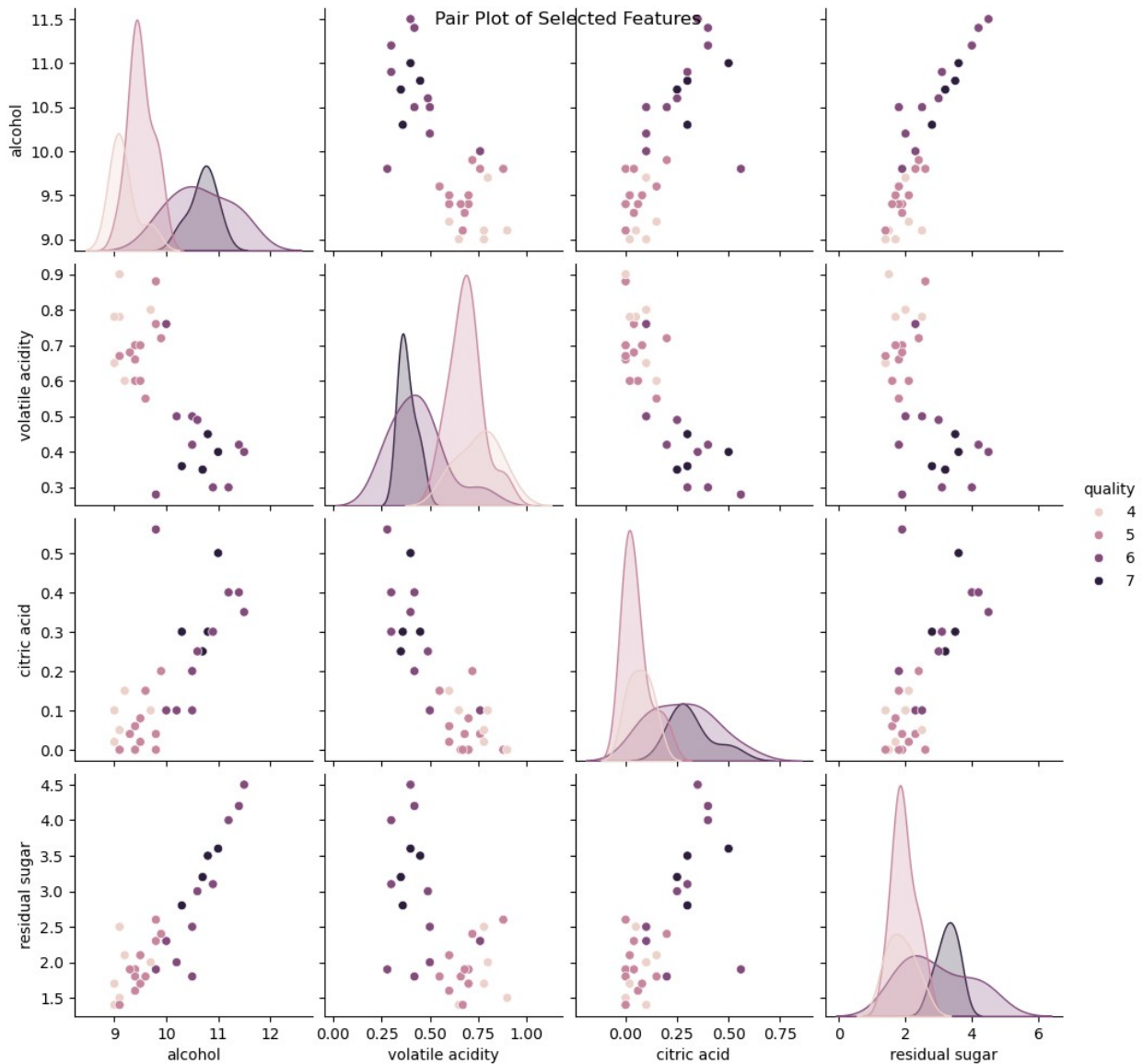
<Figure size 800x600 with 0 Axes>

<Figure size 800x600 with 0 Axes>

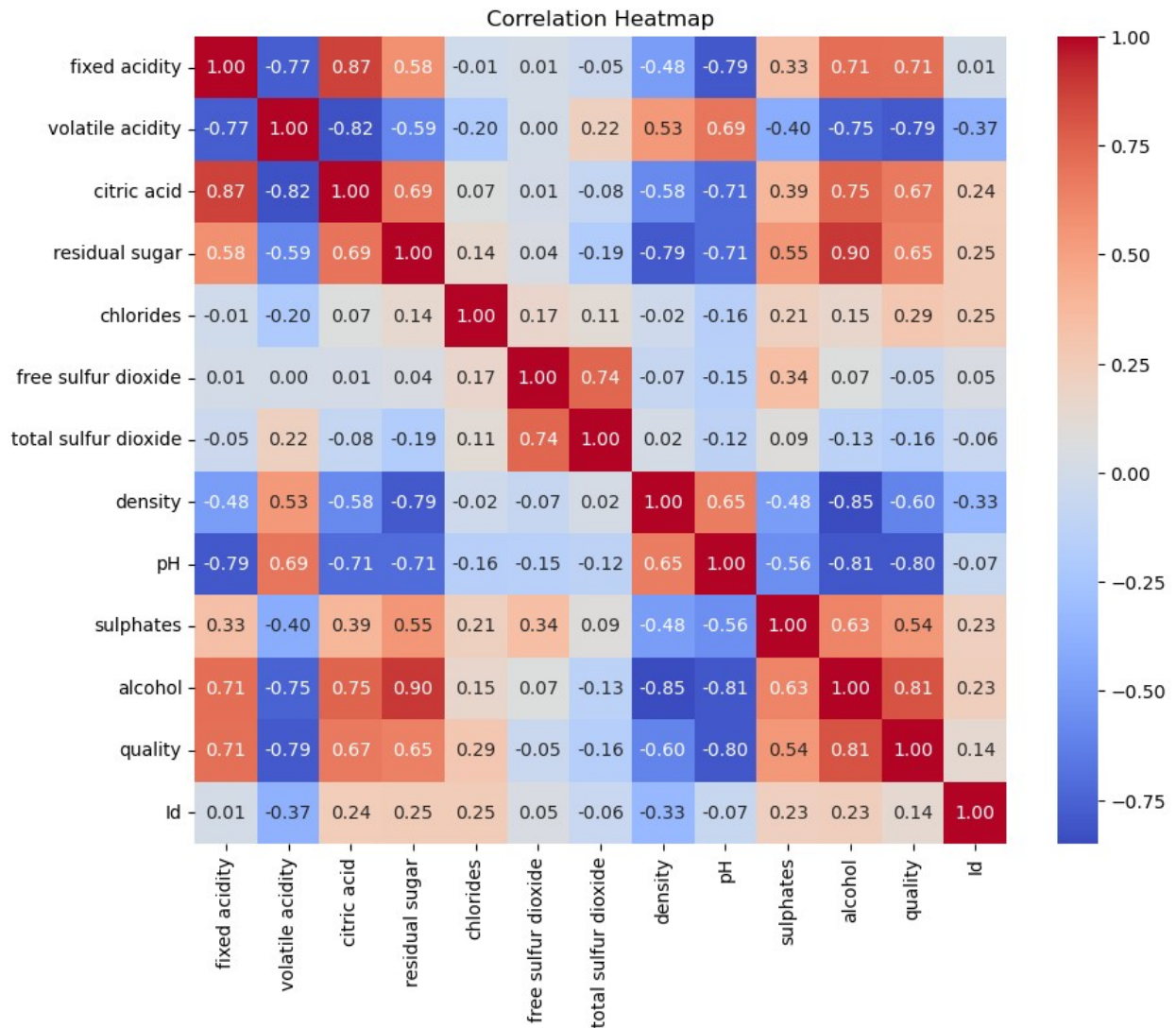
<Figure size 800x600 with 0 Axes>



```
sns.pairplot (data, vars= ['alcohol', 'volatile acidity', 'citric  
acid', 'residual sugar'], hue='quality', diag_kind='kde')  
plt.suptitle ("Pair Plot of Selected Features")  
plt.show ( )
```

```
corr_matrix = data.corr ( )
plt.figure (figsize = (10, 8))
sns.heatmap (corr_matrix, annot=True, cmap="coolwarm", fmt=".2f")
plt.title ("Correlation Heatmap")
plt.show ( )
```



```

features = ['alcohol', 'volatile acidity', 'citric acid', 'residual
sugar']
for feature in features:
    plt.figure(figsize = (6, 4))
    sns.histplot(data[data[feature]], kde=True, bins=20)
    plt.title(f"Distribution of {feature}")
    plt.show()

```

<Figure size 600x400 with 0 Axes>

<Figure size 600x400 with 0 Axes>

<Figure size 600x400 with 0 Axes>

