STUDY OF WINE QUALITY

We are investigating a dataset containing details on the physicochemical attributes of red wine. Our objective is to create comprehensive data summaries for each variable, achieved through an examination of their distributions, the detection of outliers, and the identification of potential data quality concerns.

In [17]:
```
# importing libraries
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

The data provided has a seperation value of ";" instead of "," as in any regular csv file. Therefore while reading the file we specify the seperation value ";" to read variabes in seperate columns

In [18]:
```
# Reading the data
df = pd.read_csv(r"E:\Linder_college\Statistical_Methods\Data_Sets\winequality-red.csv"
                 ,sep=';')
```

In [4]:
```
df['quality'].unique()
```

Out[4]:
```
array([5, 6, 7, 4, 8, 3], dtype=int64)
```

In [6]:
```
df.head()
```

Out[6]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5

Identifying Data Sample Size

In [86]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1599 entries, 0 to 1598
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   fixed acidity         1599 non-null   float64
 1   volatile acidity      1599 non-null   float64
 2   citric acid           1599 non-null   float64
 3   residual sugar        1599 non-null   float64
 4   chlorides             1599 non-null   float64
 5   free sulfur dioxide   1599 non-null   float64
 6   total sulfur dioxide  1599 non-null   float64
 7   density               1599 non-null   float64
 8   pH                    1599 non-null   float64
 9   sulphates             1599 non-null   float64
```

```
 10  alcohol               1599 non-null   float64
 11  quality               1599 non-null   int64
dtypes: float64(11), int64(1)
memory usage: 150.0 KB
```

- The dataset contains 1599 record and 12 variables
- Apart from the quality column all other columns are continious variables
- Quality is a discrete variable

Summarising the data and Identifying outliers

Fixed Acidity

Summarizing the data for fixed acidity

In [18]:
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['fixed acidity'].describe()
```

Out[18]:
```
count    1599.000000
mean        8.319637
std         1.741096
min         4.600000
25%         7.100000
50%         7.900000
75%         9.200000
max        15.900000
Name: fixed acidity, dtype: float64
```

In [155…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['fixed acidity'])==True])
print("Null Values: ",null_values)

median = df['fixed acidity'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  7.9
```

- For the fixed acidity column we see that the mean is 8.319637 with a standard deviation of 1.741096.
- We identify that the mean is greater than median hence we can say that data is skewed to the right side.
- We also see that there is a data quality issue as the max value is quite larger than the mean value, also we see that 75% of data has value below 9.2. Therefore the value 15.9 is an outlier and we may have outliers for data having higher values.
- In this case median would provide more appropriate value for center of data

Reporting the variablility in data

In [19]:
```
mean = df['fixed acidity'].mean()
sd = df['fixed acidity'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['fixed acidity']>(mean-2*sd))
```

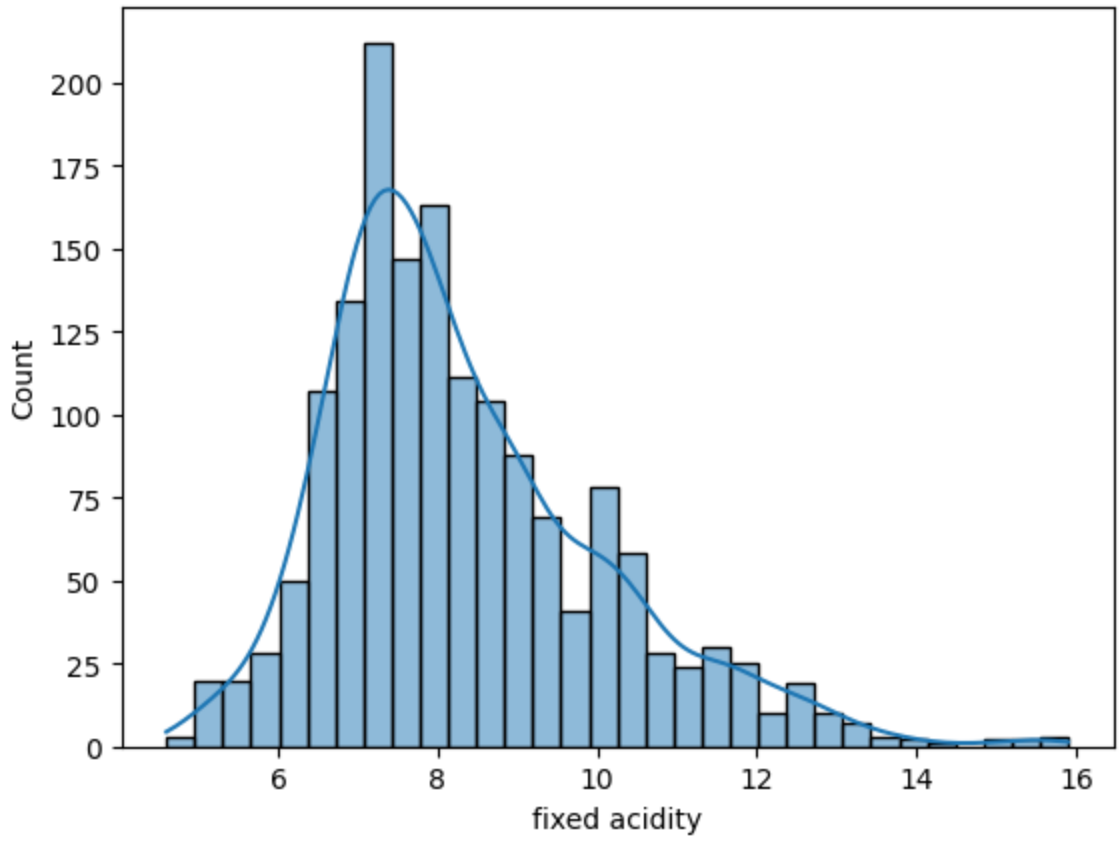
```
                                &(df['fixed acidity']<(mean+2*sd))]['fixed acidity'])


percent_values_in_range = (no_of_values_in_range/total_values)*100

value = "fixed acidity"

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for fixed acidity is  94.996873045
65353
```

Visualizing the distribution

In [23]:
```
sns.histplot(data = df,x = 'fixed acidity',kde = True)
```

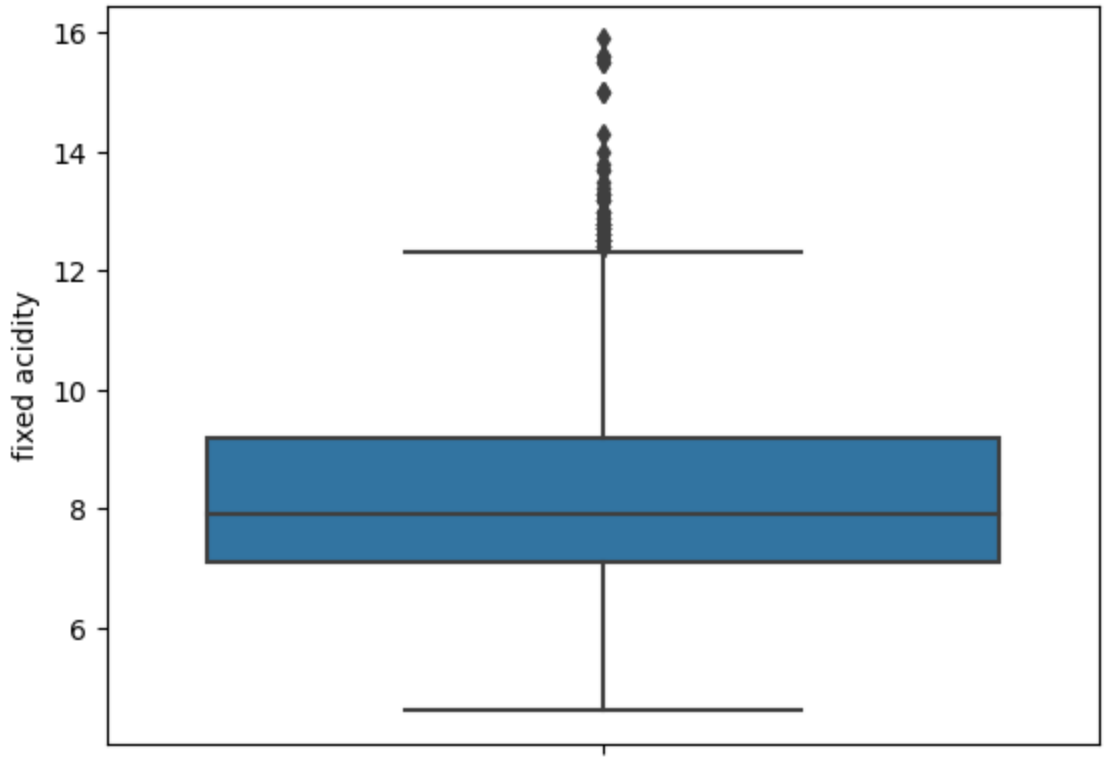
Out[23]:
```
<Axes: xlabel='fixed acidity', ylabel='Count'>
```



From the above plot we can infer that the distribution is skewed towards the right. Most of the data is normally distributed but we have a long tail towards the right

In [24]:
```
sns.boxplot(data = df,y = 'fixed acidity')
```

Out[24]:
```
<Axes: ylabel='fixed acidity'>
```



The box plots shows us that the data has outliers towards the upper end of the values.

Volatile Acidity

Summarizing the data for volatile acidity

In [88]:
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['volatile acidity'].describe()
```

Out[88]:
```
count    1599.000000
mean        0.527821
std         0.179060
min         0.120000
25%         0.390000
50%         0.520000
75%         0.640000
max         1.580000
Name: volatile acidity, dtype: float64
```

In [102…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['volatile acidity'])==True])
print("Null Values: ",null_values)

median = df['volatile acidity'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  0.52
```

- For the Volatile acidity column we see that the mean is 0.527821 with a standard deviation of 0.179060.
- We identify that the mean is slightly greater than median hence we can say that data is skewed slightly to the right side.

- We also see that there is a data quality issue as the max value is quite larger than the mean value, also we see that 75% of data has value below 0.64. Therefore the data with higher values may have outliers.

Reporting the variablility in data

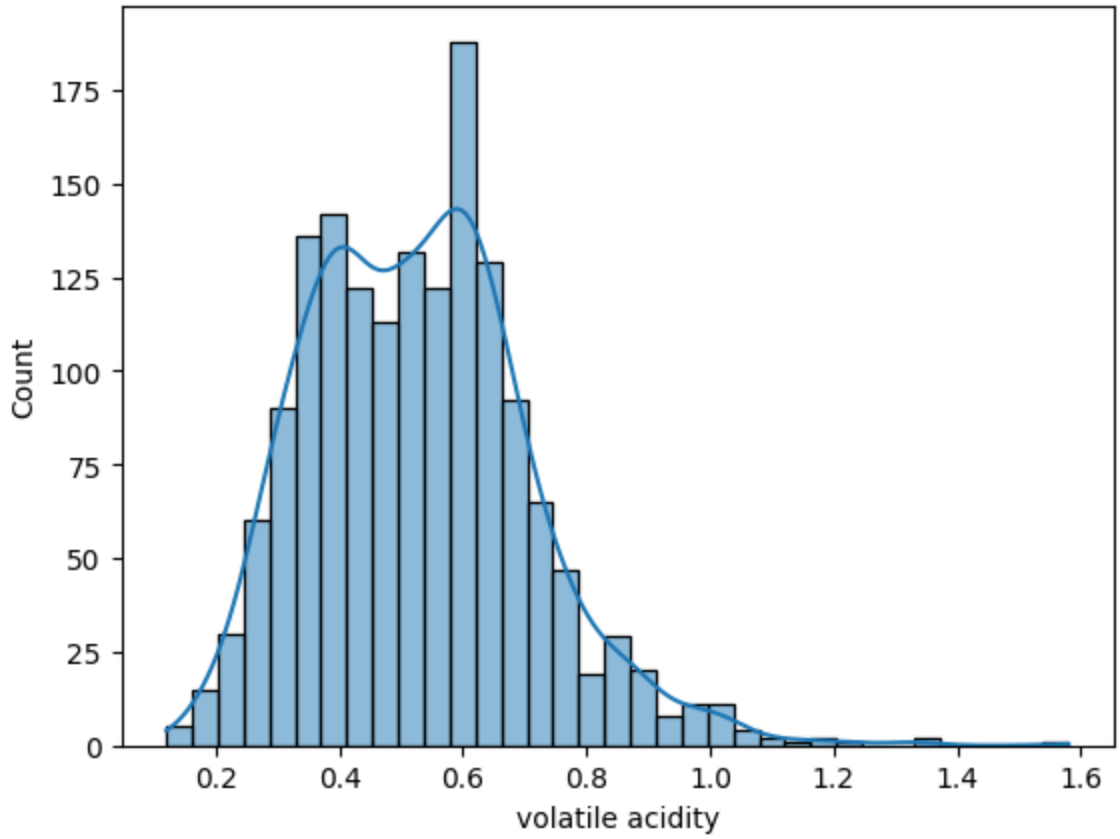
In [91]:
```
mean = df['volatile acidity'].mean()
sd = df['volatile acidity'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['volatile acidity']>(mean-2*sd))&
                               (df['volatile acidity']<(mean+2*sd))]['volatile acidity']

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = "volatile acidity"

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for volatile acidity is  96.497811
13195748
```

Visualizing the distribution

In [29]:
```
sns.histplot(data = df,x = 'volatile acidity',kde = True)
```

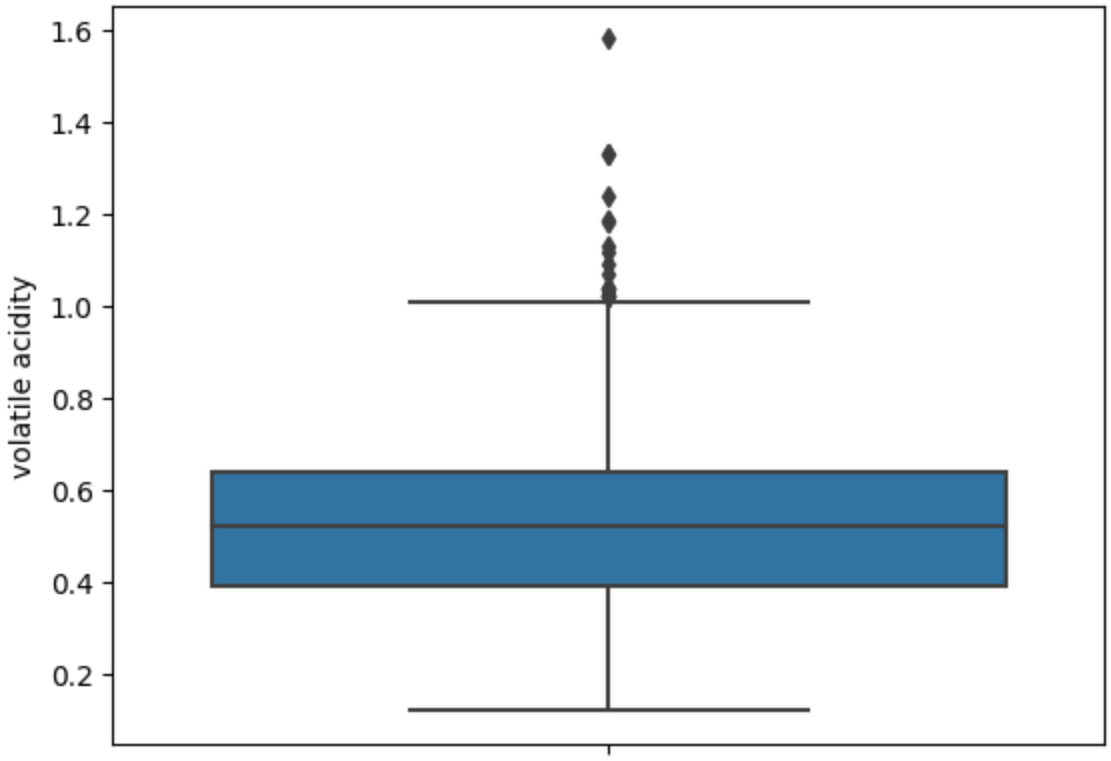
Out[29]:
```
<Axes: xlabel='volatile acidity', ylabel='Count'>
```



- From the plot we can infer that the data is skewed to the right

In [30]:
```
sns.boxplot(data = df,y = 'volatile acidity')
```

Out[30]:
```
<Axes: ylabel='volatile acidity'>
```



The box plot shows that there are outliers towards the higher end of the values

Citric Acid

Summarizing the data for citric acid

In [32]:
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['citric acid'].describe()
```

Out[32]:
```
count    1599.000000
mean        0.270976
std         0.194801
min         0.000000
25%         0.090000
50%         0.260000
75%         0.420000
max         1.000000
Name: citric acid, dtype: float64
```

In [106…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['citric acid'])==True])
print("Null Values: ",null_values)

median = df['citric acid'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  0.26
```

- For the Citric acid column we see that the mean is 0.270975 with a standard deviation of 0.194801.
- We identify that the mean is slightly greater than median hence we can say that data is skewed slightly to the right side.

- We also see that there is a data quality issue as the max value is quite larger than the mean value, also we see that 75% of data has value below 0.42. Therefore the data at higher values has outliers.

Reporting the variability in data

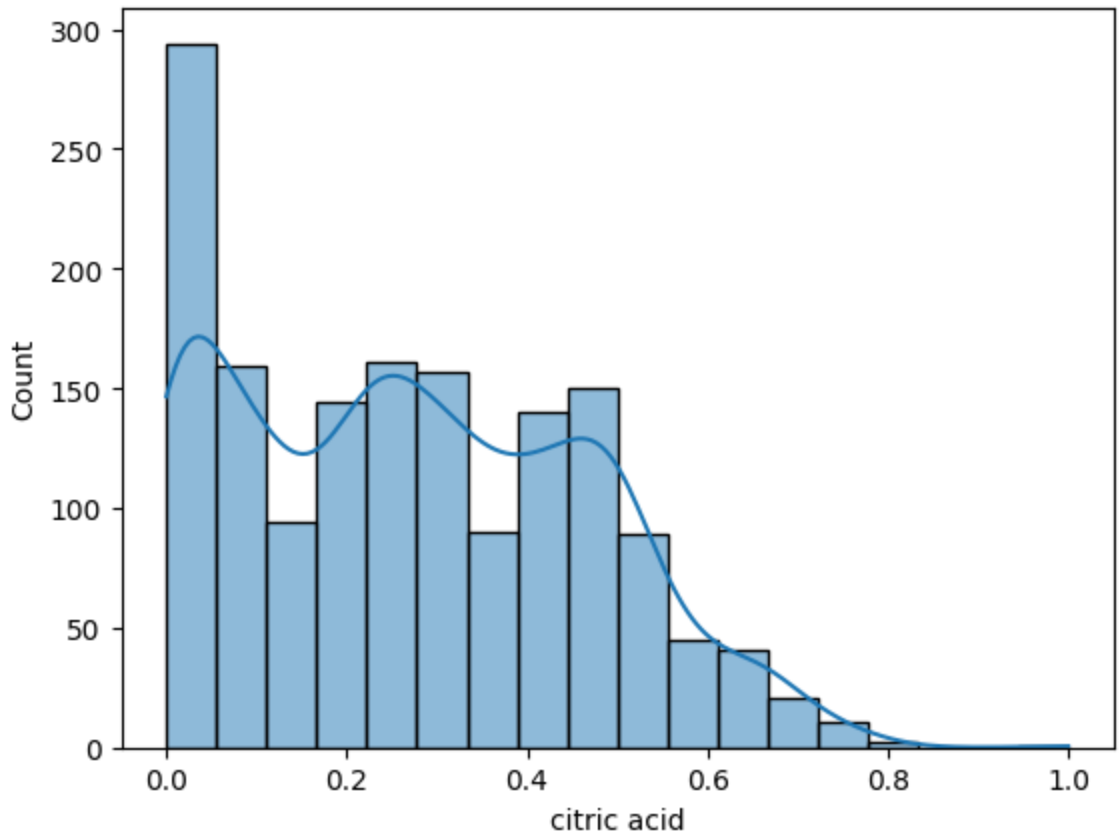
In [108…]
```
mean = df['citric acid'].mean()
sd = df['citric acid'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['citric acid']>(mean-2*sd))&
                               (df['citric acid']<(mean+2*sd))]['citric acid'])

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = 'citric acid'

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for citric acid is  97.81113195747
342
```

Visualizing the distribution

In [36]:
```
sns.histplot(data = df,x = 'citric acid',kde = True)
```

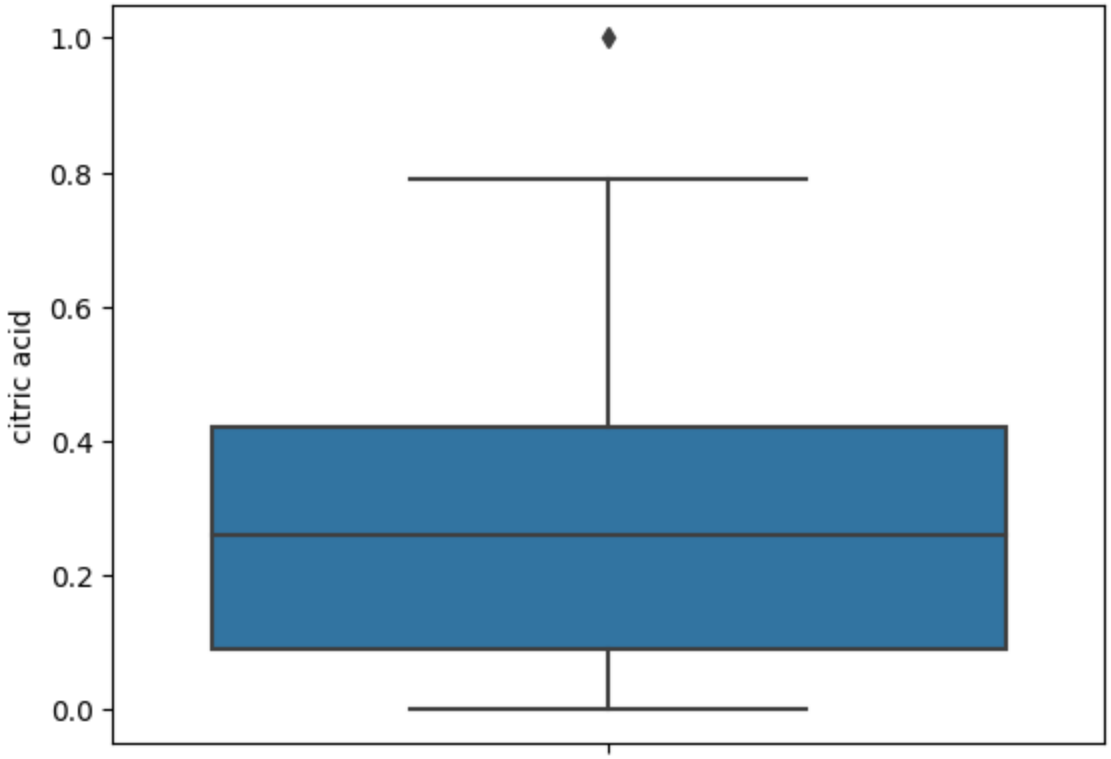
Out[36]:
```
<Axes: xlabel='citric acid', ylabel='Count'>
```



From the plot we can infer that the data is skewed towards right. The distribution has a longer tail towards the right.

In [37]:
```
sns.boxplot(data = df,y = 'citric acid')
```

Out[37]:
```
<Axes: ylabel='citric acid'>
```



The box plot shows that there are outliers towards the higher end of the values

Residual Sugar

Summarizing the data for Residual Sugar

In [110…
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['residual sugar'].describe()
```

Out[110]:
```
count    1599.000000
mean        2.538806
std         1.409928
min         0.900000
25%         1.900000
50%         2.200000
75%         2.600000
max        15.500000
Name: residual sugar, dtype: float64
```

In [151…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['residual sugar'])==True])
print("Null Values: ",null_values)

median = df['residual sugar'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  2.2
```

- From the data for this variable we see that the mean is 2.538806 with a standard deviation of 1.409928.
- We identify that the mean is slightly higher than median hence we can say that data is skewed slightly to the right side.

- We also see that there is a data quality issue as the max value is quite larger than the mean value, also we see that 75% of data has value below 2.600. Therefore the data with higher values may have outliers.

Finding the Variability of data

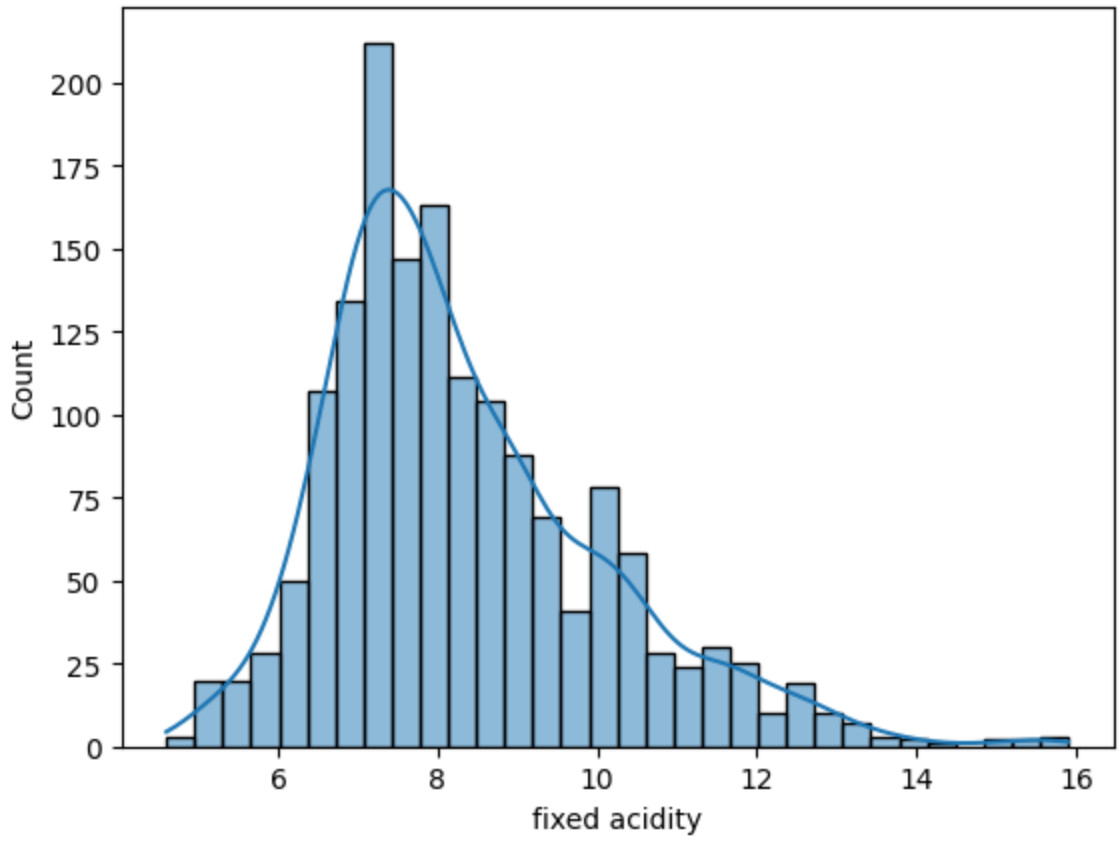
In [152…
```
mean = df['residual sugar'].mean()
sd = df['residual sugar'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['residual sugar']>(mean-2*sd))
                               &(df['residual sugar']<(mean+2*sd))]['residual sugar'])

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = 'residual sugar'

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for residual sugar is  95.30956848
030019
```

Visualizing the distribution

In [153…
```
sns.histplot(data = df,x = 'fixed acidity',kde = True)
```

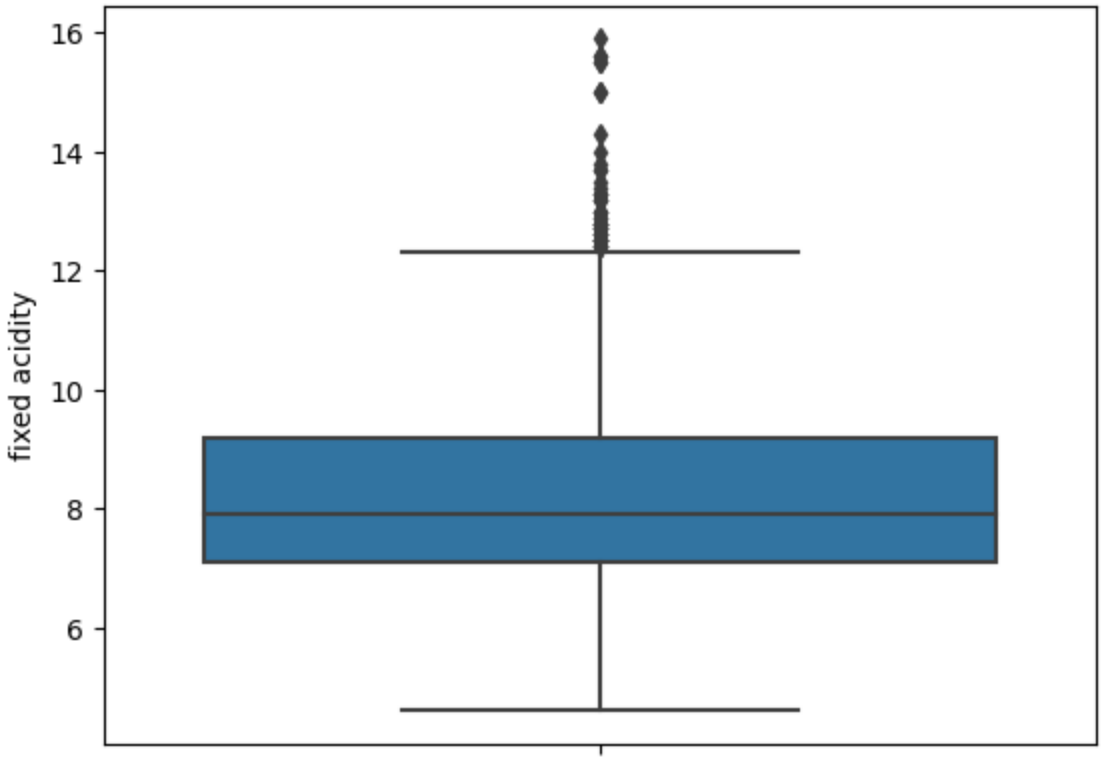
Out[153]:
```
<Axes: xlabel='fixed acidity', ylabel='Count'>
```



From the above plot we can infer that the distribution is skewed towards the right as we have outliers towards higher end values

In [42]:
```
sns.boxplot(data = df,y = 'fixed acidity')
```

Out[42]:
```
<Axes: ylabel='fixed acidity'>
```



Through the box plot we can infer that the there are outliers in data at higher values and hence we see a skew in the distribution towards the right

Chlorides

Summarizing the data for chlorides

In [147…
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['chlorides'].describe()
```

Out[147]:
```
count    1599.000000
mean        0.087467
std         0.047065
min         0.012000
25%         0.070000
50%         0.079000
75%         0.090000
max         0.611000
Name: chlorides, dtype: float64
```

In [148…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['chlorides'])==True])
print("Null Values: ",null_values)

median = df['chlorides'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  0.079
```

- From the data for this variable we see that the mean is 0.087467 with a standard deviation of 0.047065.
- We identify that the mean is greater than median hence we can say that data is skewed to the right side.

- We also see that there is a data quality issue as the max value is quite larger than the mean value, also we see that 75% of data has value below 0.090000. Therefore the data with higher values may have outliers.

Finding the Variability of data

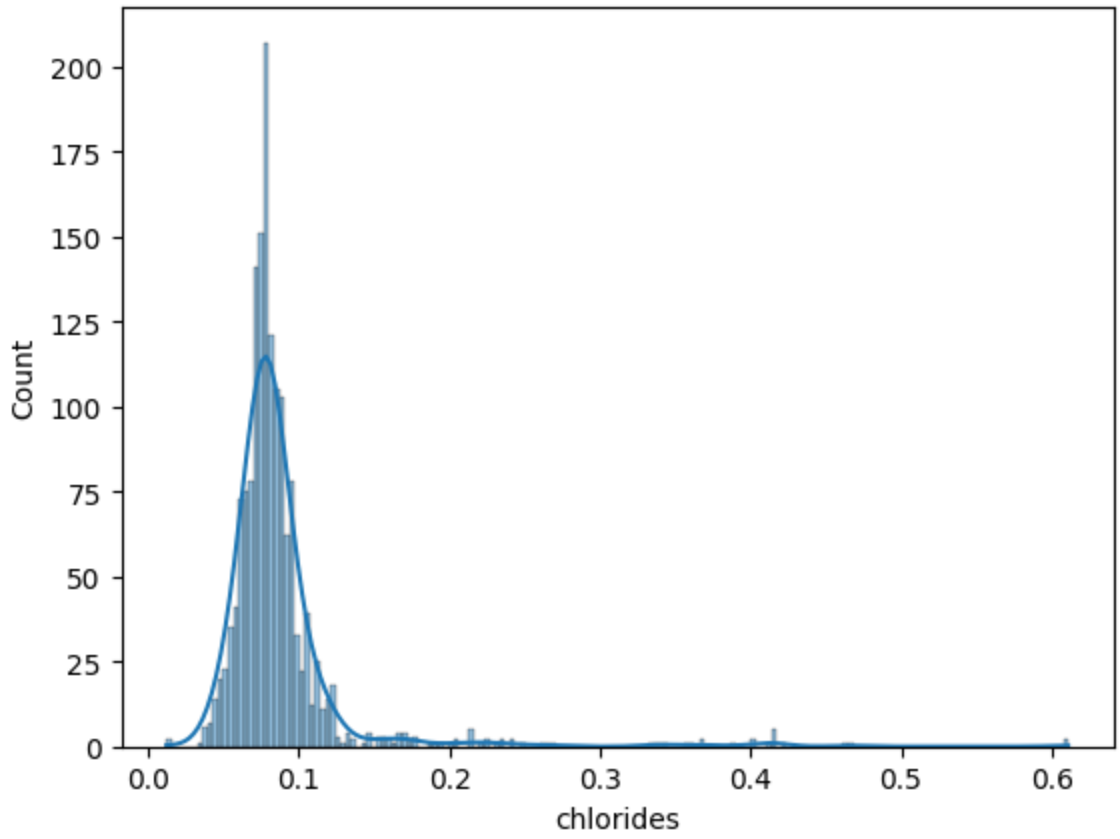
In [149…
```
mean = df['chlorides'].mean()
sd = df['chlorides'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['chlorides']>(mean-2*sd))
                               &(df['chlorides']<(mean+2*sd))]['chlorides'])

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = 'chlorides'

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for chlorides is  97.18574108818011
```

Visualizing the distribution

In [150…
```
sns.histplot(data = df,x = 'chlorides',kde = True)
```

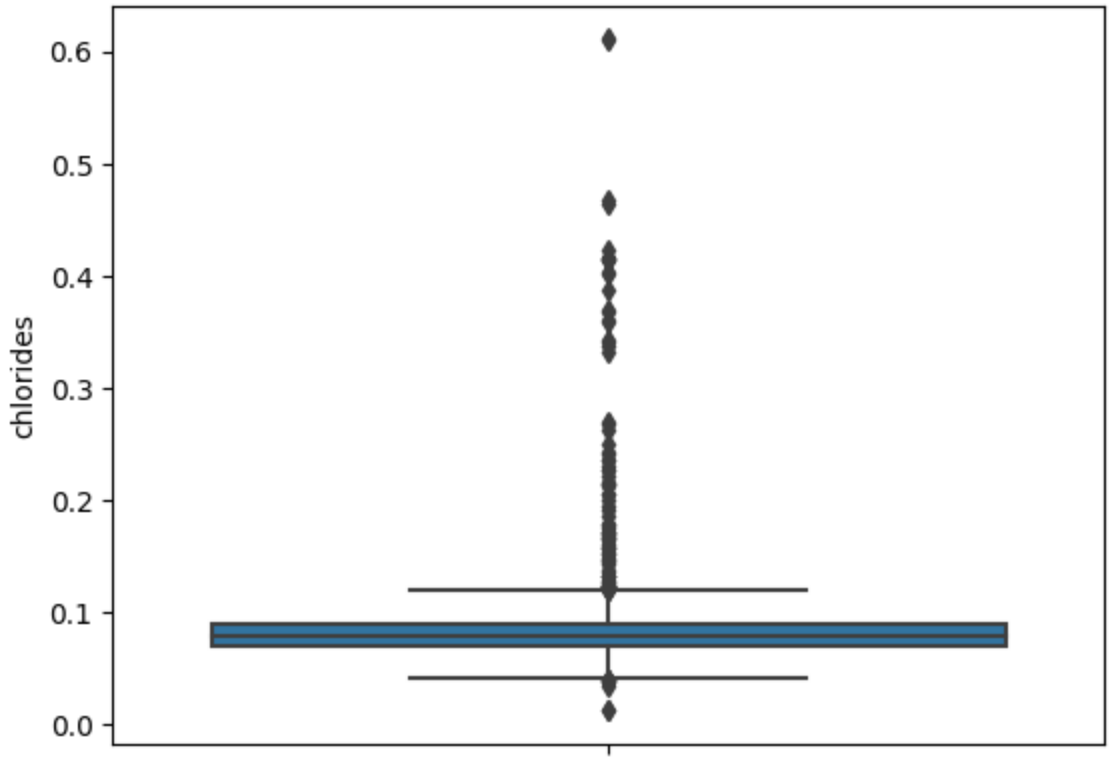
Out[150]:
```
<Axes: xlabel='chlorides', ylabel='Count'>
```



We can infer from the plot that the distribution is skewed towards the right

In [47]:
```
sns.boxplot(data = df,y = 'chlorides')
```

Out[47]:
```
<Axes: ylabel='chlorides'>
```



Through the box plot we can infer that the there are high number of outliers in data at higher values and hence we see a skew in the distribution towards the right

Free sulphur dioxide

Summarizing the data for free sulphur dioxide

In [143…
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['free sulfur dioxide'].describe()
```

Out[143]:
```
count    1599.000000
mean       15.874922
std        10.460157
min         1.000000
25%         7.000000
50%        14.000000
75%        21.000000
max        72.000000
Name: free sulfur dioxide, dtype: float64
```

In [144…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['free sulfur dioxide'])==True])
print("Null Values: ",null_values)

median = df['free sulfur dioxide'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  14.0
```

- From the data for this variable we see that the mean is 15.874922 with a standard deviation of 10.460157.
- We identify that the mean is higher than median hence we can say that data is skewed towards the right side.

- We also see that there is a data quality issue as the max value is quite larger than the mean value, also we see that 75% of data has value below 21. Therefore the data with higher values may have outliers.

Finding the Variability of data

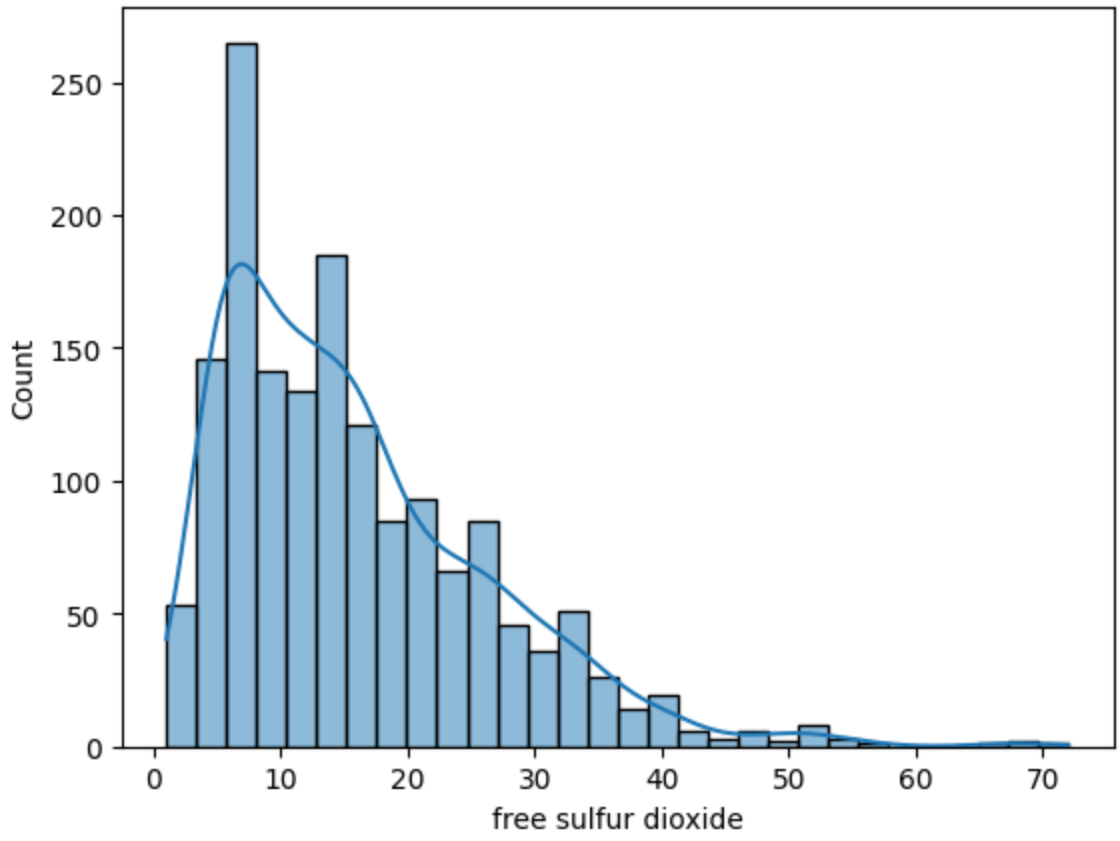
In [145…
```
mean = df['free sulfur dioxide'].mean()
sd = df['free sulfur dioxide'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['free sulfur dioxide']>(mean-2*sd))
                               &(df['free sulfur dioxide']<(mean+2*sd))]['free sulfur di

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = 'free sulfur dioxide'

print(f"Percentage of values within two std deviation of mean for {value} is ",
      percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for free sulfur dioxide is  95.872
42026266416
```

Visualizing the distribution

In [146…
```
sns.histplot(data = df,x = 'free sulfur dioxide',kde = True)
```

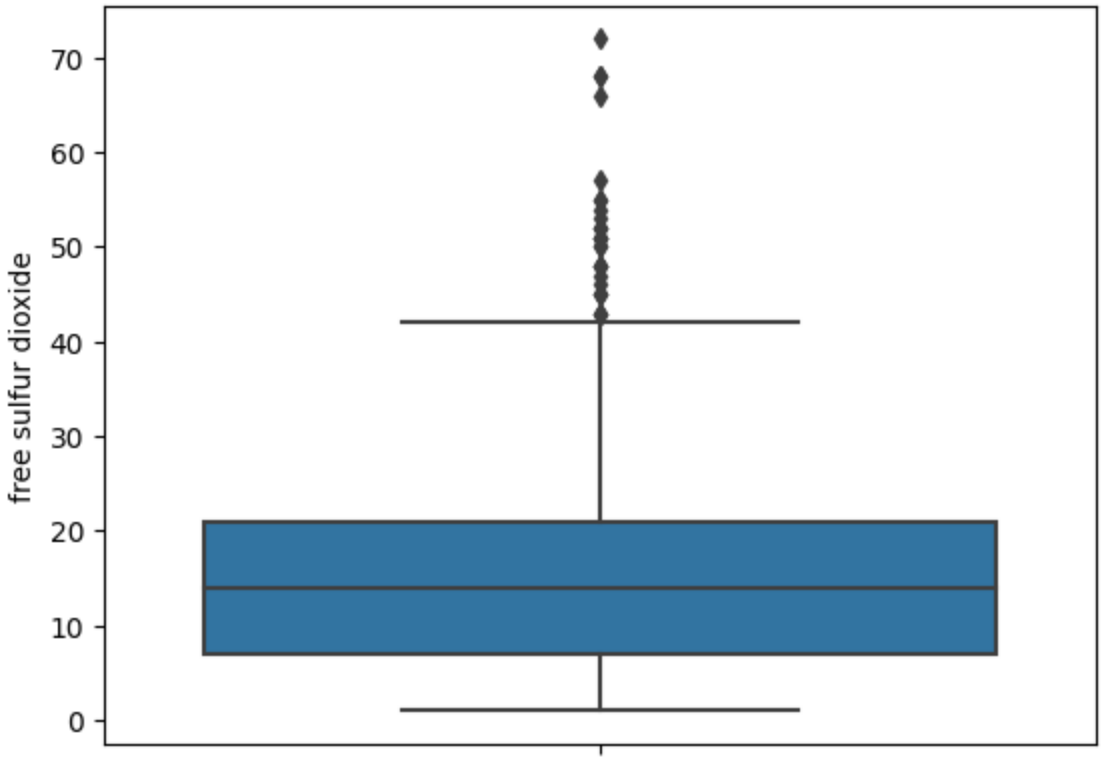
Out[146]:
```
<Axes: xlabel='free sulfur dioxide', ylabel='Count'>
```



From the above plot we can infer that the distribution is skewed towards the right.

In [53]:
```
sns.boxplot(data = df,y = 'free sulfur dioxide')
```

Out[53]:
```
<Axes: ylabel='free sulfur dioxide'>
```



Through the box plot we can infer that the there are outliers in data at higher values and hence we see a skew in the distribution towards the right

Total Sulphur dioxide

Summarizing the data for volatile acidity

In [135…
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['total sulfur dioxide'].describe()
```

Out[135]:
```
count    1599.000000
mean       46.467792
std        32.895324
min         6.000000
25%        22.000000
50%        38.000000
75%        62.000000
max       289.000000
Name: total sulfur dioxide, dtype: float64
```

In [137…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['total sulfur dioxide'])==True])
print("Null Values: ",null_values)

median = df['total sulfur dioxide'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  38.0
```

- From the data for this variable we see that the mean is 46.467792 with a standard deviation of 32.895324.
- We identify that the mean is quite higher than median hence we can say that data is skewed to the right side.

- We also see that there is a data quality issue as the max value is quite larger than the mean value, also we see that 75% of data has value below 62. Therefore the data with higher values has outliers.

Finding the Variability of data

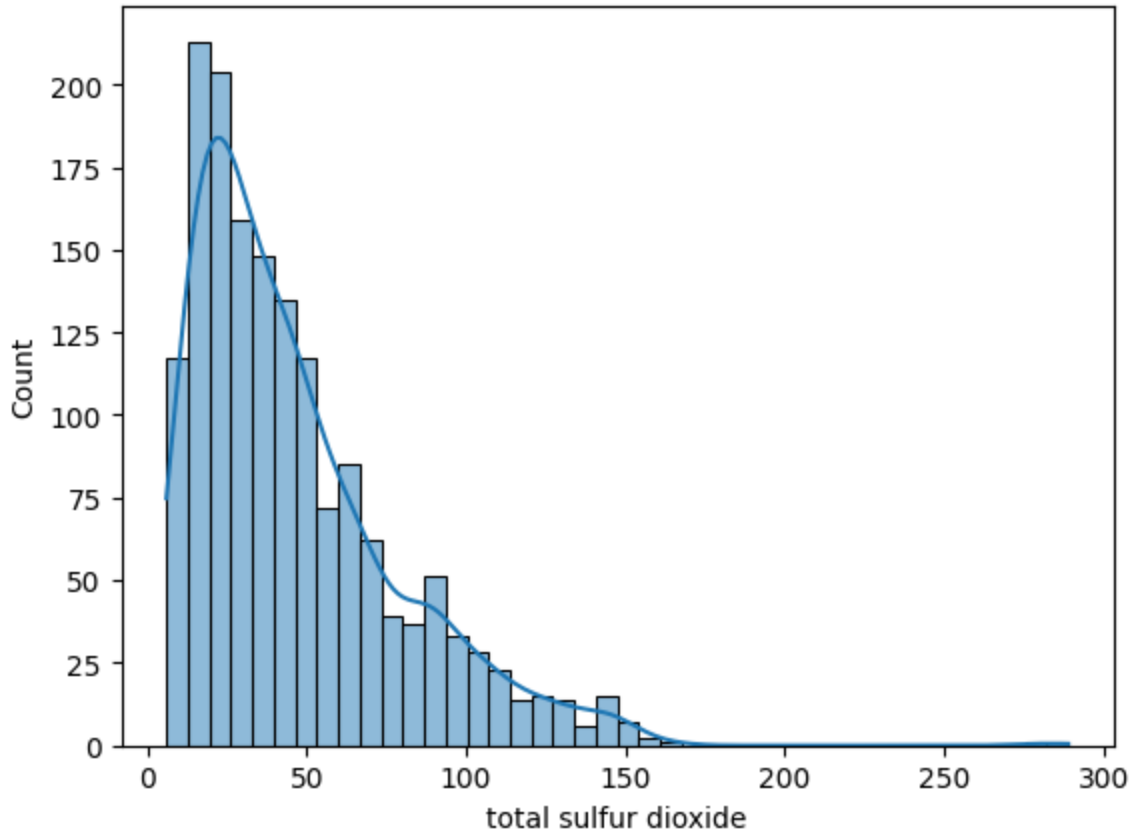
In [139…
```
mean = df['total sulfur dioxide'].mean()
sd = df['total sulfur dioxide'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['total sulfur dioxide']>(mean-2*sd))
                               &(df['total sulfur dioxide']<(mean+2*sd))]['total sulfur

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = 'total sulfur dioxide'

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for total sulfur dioxide is  94.99
687304565353
```

In [ ]:
```
### Visualizing the distribution
```

In [140…
```
sns.histplot(data = df,x = 'total sulfur dioxide',kde = True)
```

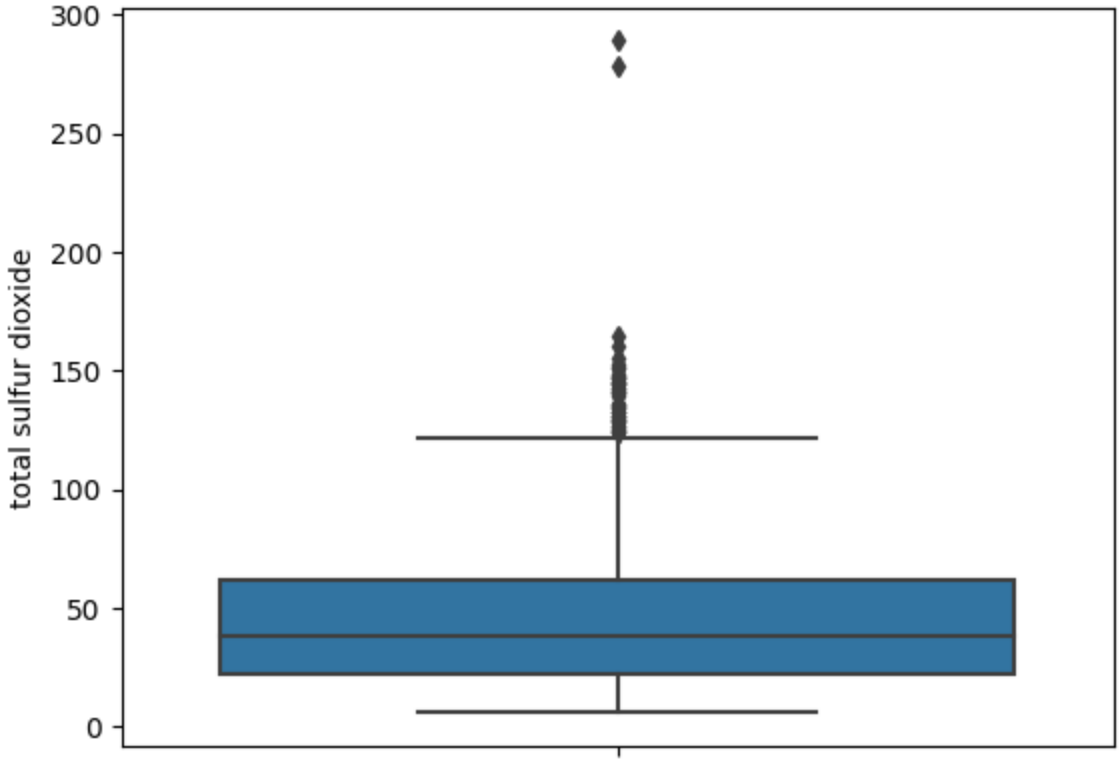
Out[140]:
```
<Axes: xlabel='total sulfur dioxide', ylabel='Count'>
```



From the plot we can see that the distribution is skewed towards the right.

In [60]:
```
sns.boxplot(data = df,y = 'total sulfur dioxide')
```

Out[60]:
```
<Axes: ylabel='total sulfur dioxide'>
```



From the box plot we can observe that there are outliers towards the higher values and hence the distribution is skewed towards the right

Density

Summarizing the data for Density

In [62]:
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['density'].describe()
```

Out[62]:
```
count    1599.000000
mean        0.996747
std         0.001887
min         0.990070
25%         0.995600
50%         0.996750
75%         0.997835
max         1.003690
Name: density, dtype: float64
```

In [131…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['density'])==True])
print("Null Values: ",null_values)

median = df['density'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  0.99675
```

- From the data for this variable we see that the mean is 0.996747 with a standard deviation of 0.001887.
- We identify that the mean is almost equal to the median hence we can say that data is normally distributed.

- We dont see any data quality issue as the data is fairly distributed normally.

Finding the Variability of data

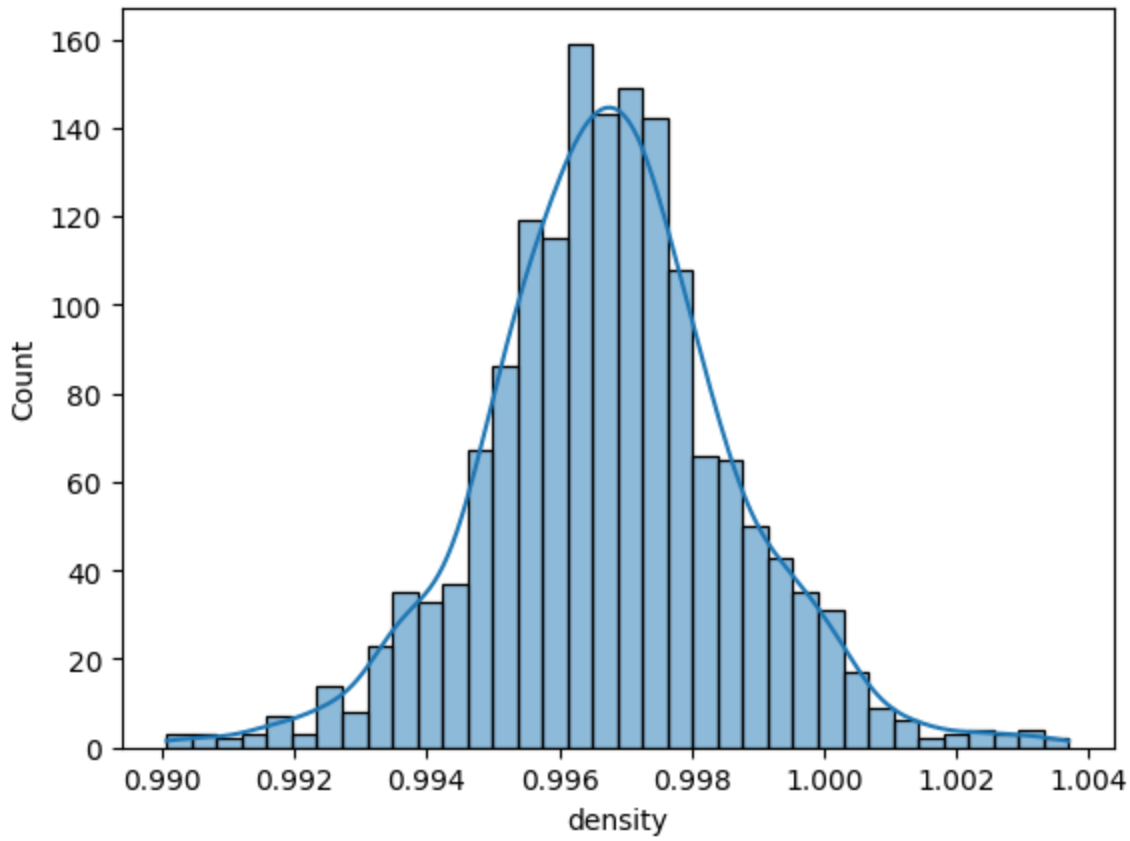
In [132…
```
mean = df['density'].mean()
sd = df['density'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['density']>(mean-2*sd))
                               &(df['density']<(mean+2*sd))]['density'])

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = 'density'

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for density is  94.93433395872421
```

Visualizing the distribution

In [64]:
```
sns.histplot(data = df,x = 'density',kde = True)
```

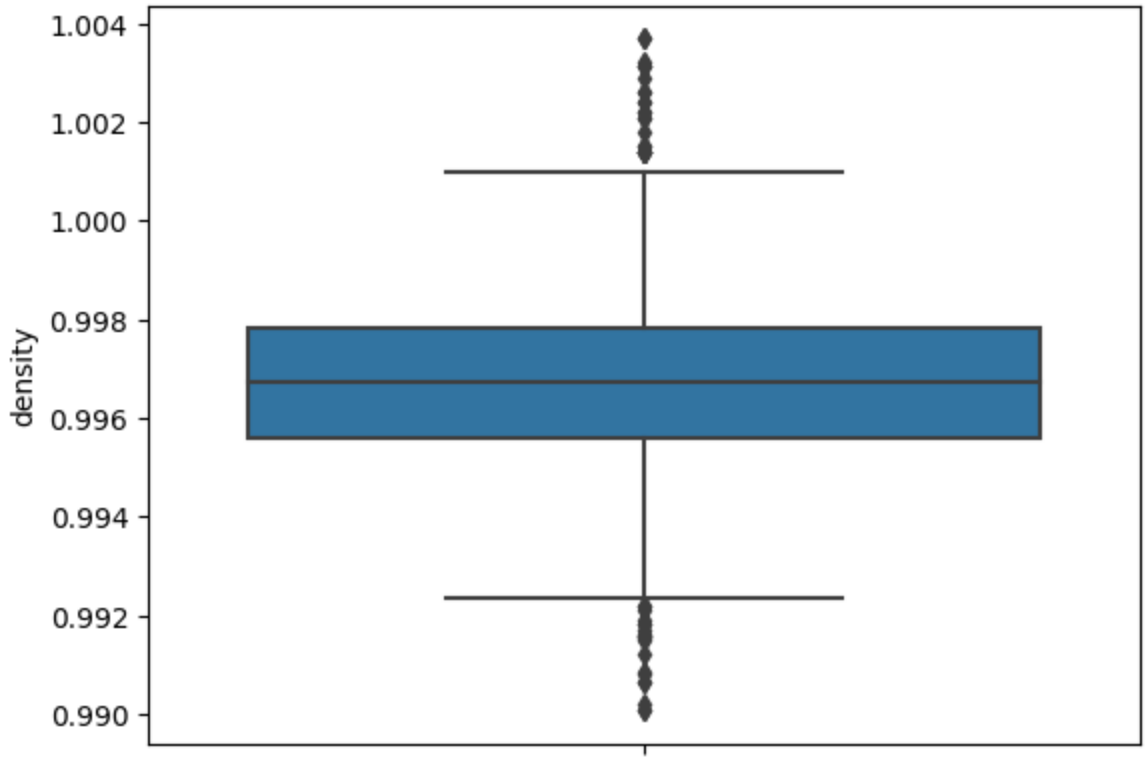
Out[64]:
```
<Axes: xlabel='density', ylabel='Count'>
```



We can infer from the plot that the data is normally distributed

In [133…
```
sns.boxplot(data = df,y = 'density')
```

Out[133]:
```
<Axes: ylabel='density'>
```



Through the box plot we can infer that the there are outliers in data at higher values as well as lower values but most of the data is normally distributed

PH

Summarizing the data for PH

In [67]:
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['pH'].describe()
```

Out[67]:
```
count    1599.000000
mean        3.311113
std         0.154386
min         2.740000
25%         3.210000
50%         3.310000
75%         3.400000
max         4.010000
Name: pH, dtype: float64
```

In [119…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['pH'])==True])
print("Null Values: ",null_values)

median = df['pH'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  3.31
```

- From the data for this variable we see that the mean is 3.311113 with a standard deviation of 0.154386.
- We identify that the mean is equal to median hence we can say that data is normally distributed.
- We see no issues with the data quality for this feature

Finding the Variability of data

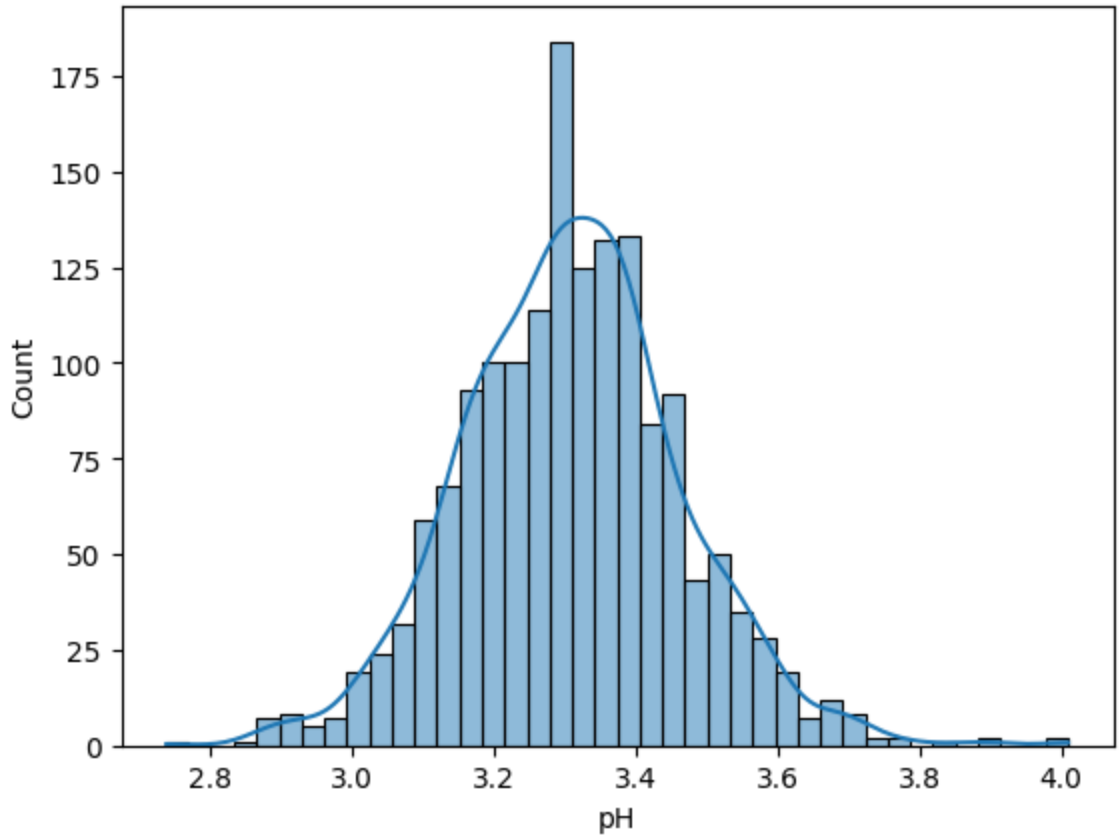
In [120…
```
mean = df['pH'].mean()
sd = df['pH'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['pH']>(mean-2*sd))
                               &(df['pH']<(mean+2*sd))]['pH'])

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = 'pH'

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for pH is  95.30956848030019
```

Visualizing the distribution

In [121…
```
sns.histplot(data = df,x = 'pH',kde = True)
```

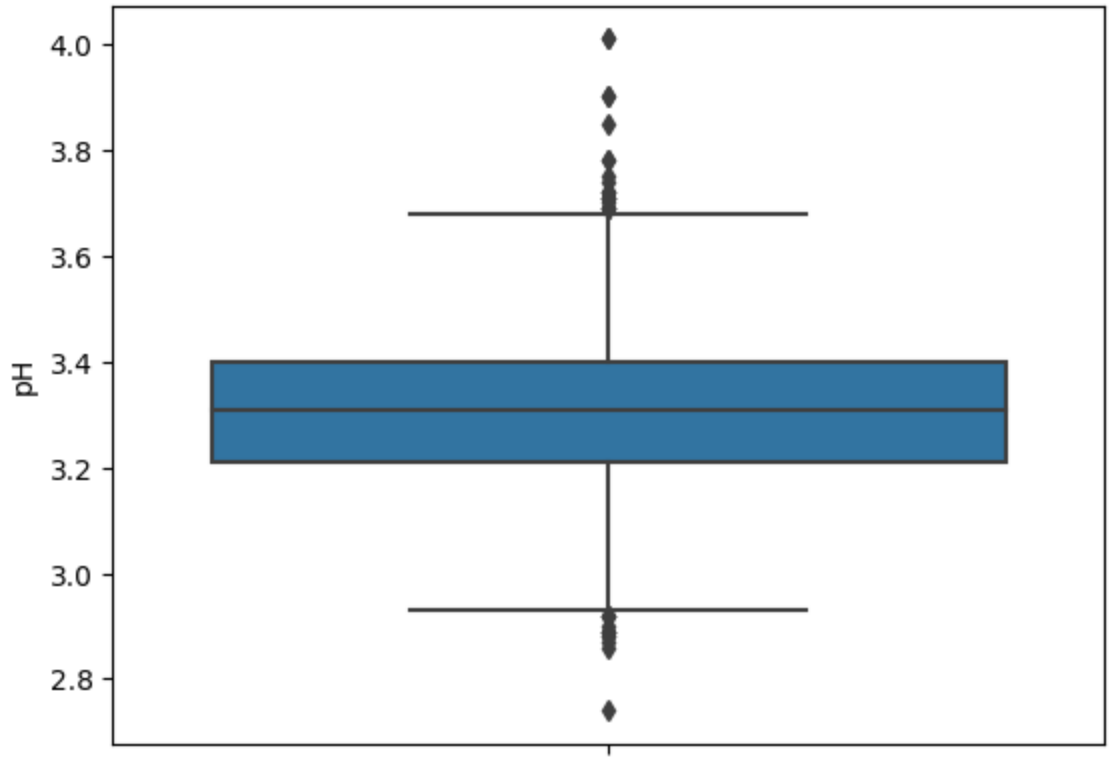
Out[121]:
```
<Axes: xlabel='pH', ylabel='Count'>
```



We can infer from the plot that the data is normally distributed but we do see outliers to the right side of the plot

In [122…
```
sns.boxplot(data = df,y = 'pH')
```

Out[122]:
```
<Axes: ylabel='pH'>
```



Through the box plot we can infer that the there are outliers in data at higher values as well as lower values but overall the majority of data is fairly normally distributed

Sulphates

Summarizing the data for sulphates

In [72]:
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['sulphates'].describe()
```

Out[72]:
```
count    1599.000000
mean        0.658149
std         0.169507
min         0.330000
25%         0.550000
50%         0.620000
75%         0.730000
max         2.000000
Name: sulphates, dtype: float64
```

In [125…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['sulphates'])==True])
print("Null Values: ",null_values)

median = df['sulphates'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  0.62
```

- From the data for this variable we see that the mean is 2.538806 with a standard deviation of 1.409928.
- We identify that the mean is slightly higher than median hence we can say that data is skewed slightly to the right side.

- We also see that there is a data quality issue as the max value is quite larger than the mean value, also we see that 75% of data has value below 0.730000. Therefore the data with higher values may have outliers.

Finding the Variability of data

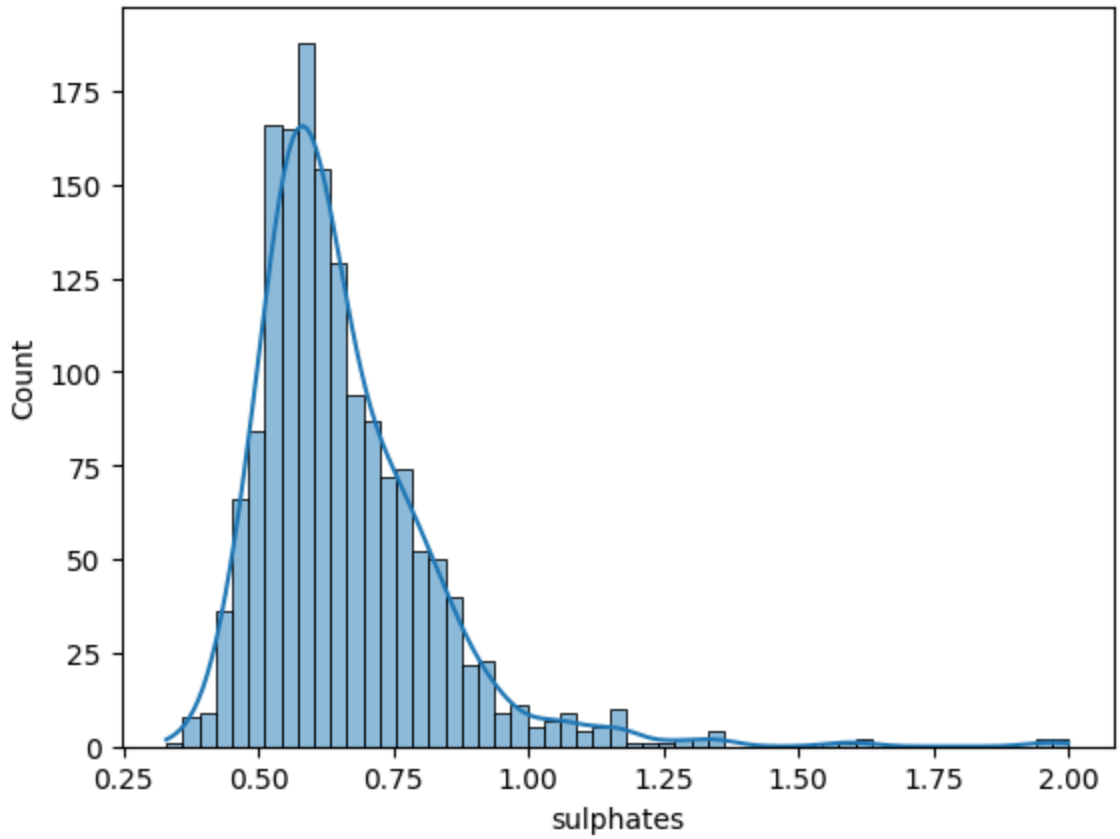
In [127…
```
mean = df['sulphates'].mean()
sd = df['sulphates'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['sulphates']>(mean-2*sd))
                               &(df['sulphates']<(mean+2*sd))]['sulphates'])

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = 'sulphates'

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for sulphates is  96.31019387116947
```

Visualizing the distribution

In [128…
```
sns.histplot(data = df,x = 'sulphates',kde = True)
```

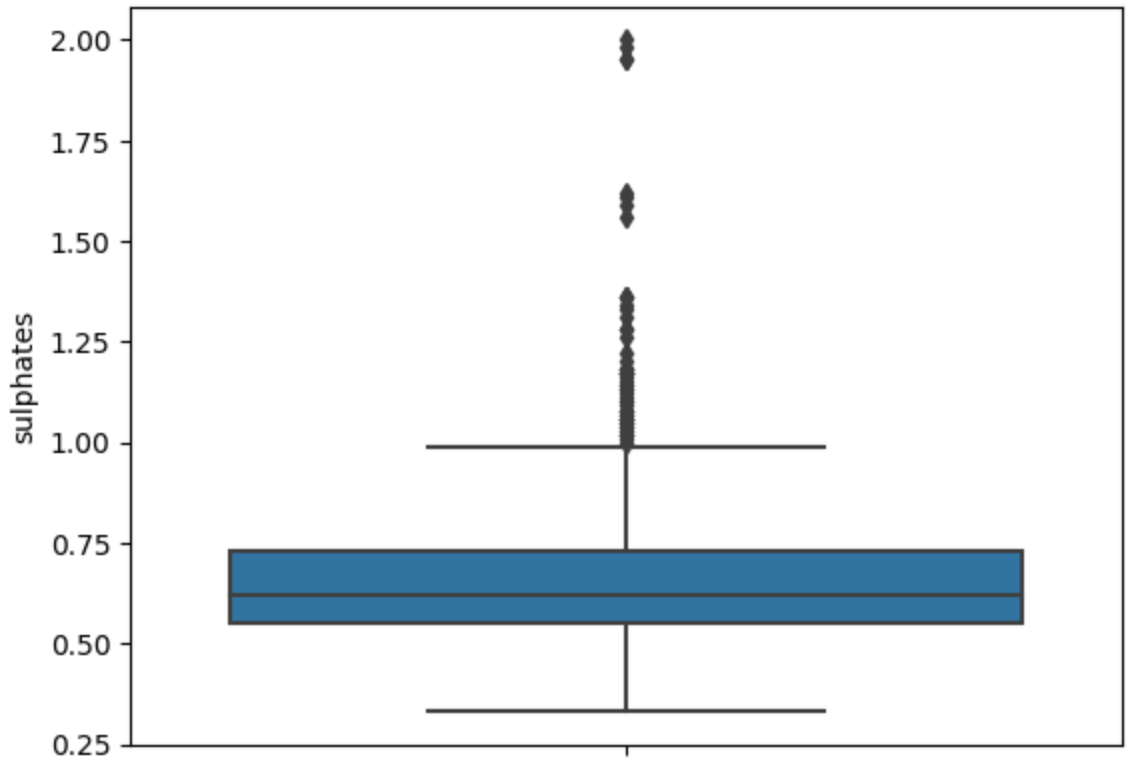
Out[128]:
```
<Axes: xlabel='sulphates', ylabel='Count'>
```



From the above plot we can infer that the data is skewed towards right. The curve towards the right has a bigger tail.

In [75]:
```
sns.boxplot(data = df,y = 'sulphates')
```

Out[75]: <Axes: ylabel='sulphates'>



Through the box plot we can infer that the there are outliers in data at higher values and hence we see a skew in the distribution towards the right

Alcohol

Summarizing the data for Alcohol

In [77]:
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['alcohol'].describe()
```

Out[77]:
```
count    1599.000000
mean       10.422983
std         1.065668
min         8.400000
25%         9.500000
50%        10.200000
75%        11.100000
max        14.900000
Name: alcohol, dtype: float64
```

In [114…
```
# Finding NUll Values
null_values = len(df[pd.isna(df['alcohol'])==True])
print("Null Values: ",null_values)

median = df['alcohol'].median()
print("Median: ",median)
```

```
Null Values:  0
Median:  10.2
```

- From the data for this variable we see that the mean is 10.422983 with a standard deviation of 1.065668.
- We identify that the mean is higher than median hence we can say that data is skewed to the right side.

Finding the Variability of data

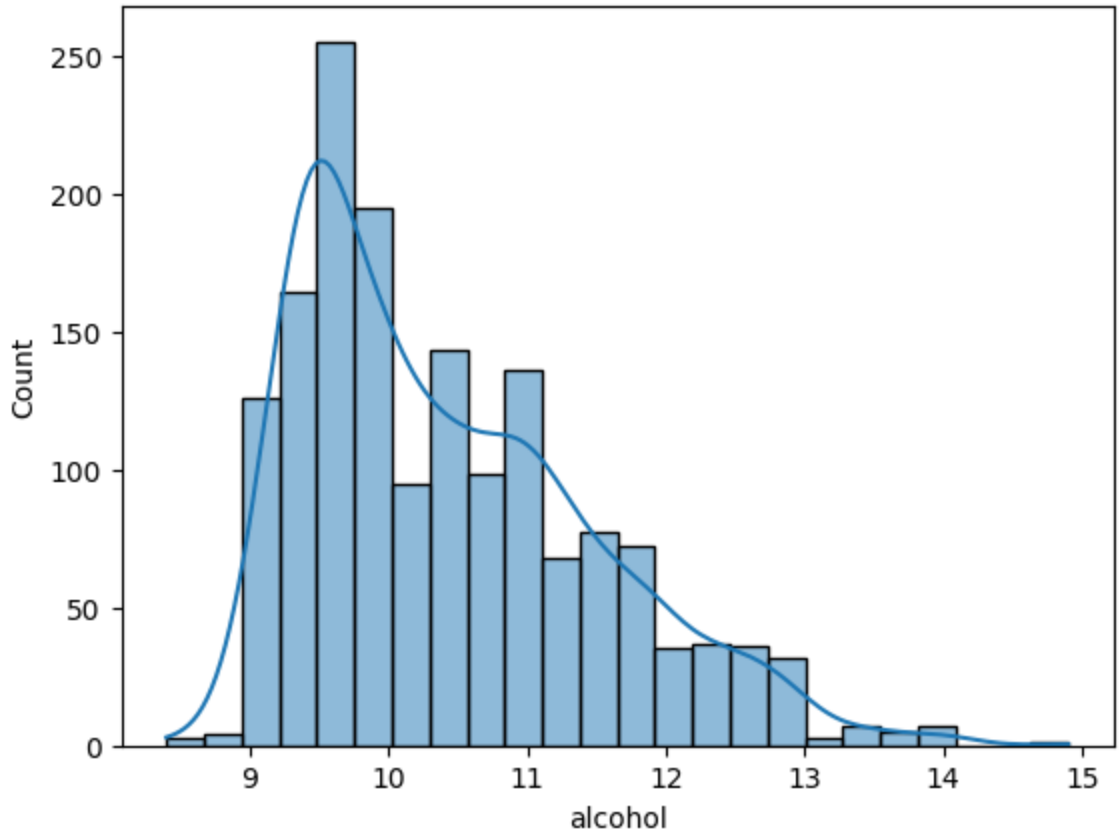
In [115…
```
mean = df['alcohol'].mean()
sd = df['alcohol'].std()
total_values = len(df)

no_of_values_in_range = len(df[(df['alcohol']>(mean-2*sd))
                               &(df['alcohol']<(mean+2*sd))]['alcohol'])

percent_values_in_range = (no_of_values_in_range/total_values)*100

value = 'alcohol'

print(f"Percentage of values within two std deviation of mean for {value} is "
      ,percent_values_in_range)
```

```
Percentage of values within two std deviation of mean for alcohol is  95.62226391494684
```

Visualizing the distribution

In [116…
```
sns.histplot(data = df,x = 'alcohol',kde = True)
```

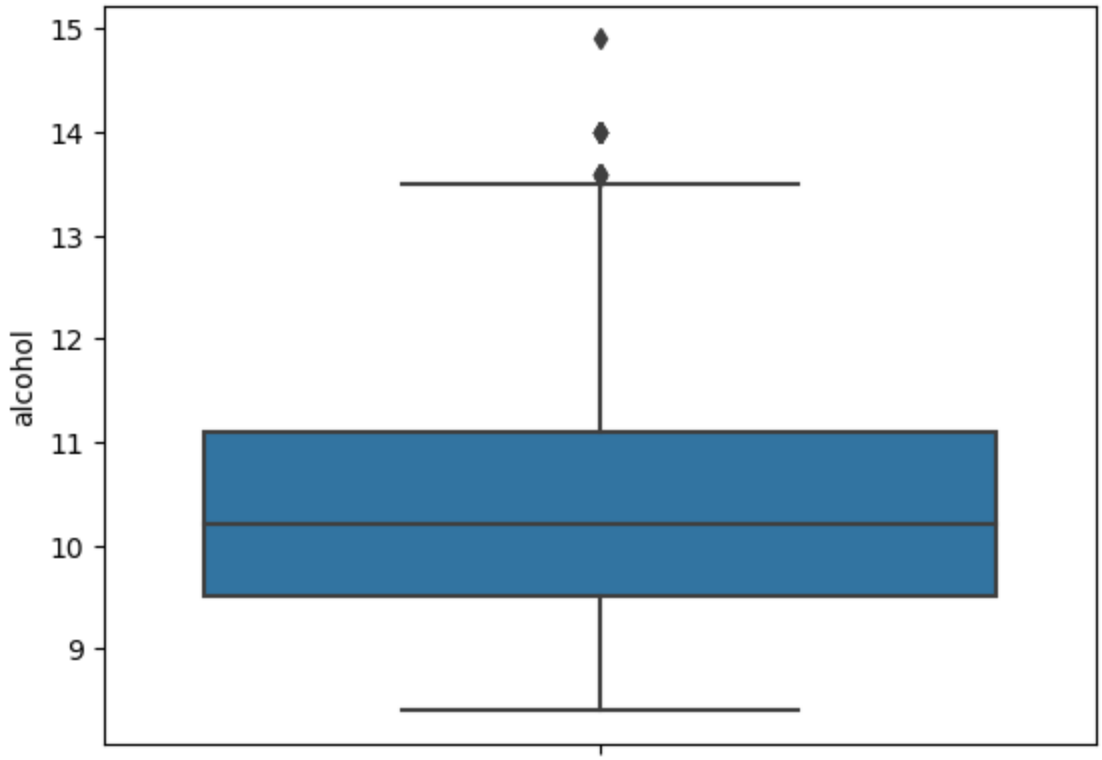
Out[116]:
```
<Axes: xlabel='alcohol', ylabel='Count'>
```



From the above plot we can infer that the data is skewed to the right

In [80]:
```
sns.boxplot(data = df,y = 'alcohol')
```

Out[80]:
```
<Axes: ylabel='alcohol'>
```



Through the box plot we can infer that the there are outliers in data at higher values and hence we see a skew in the distribution towards the right

Quality

Summarizing the data for Quality

In [82]:
```
#Checking the mean, median, standard deviation and quartile values of the variable

df['quality'].describe()
```

Out[82]:
```
count    1599.000000
mean        5.636023
std         0.807569
min         3.000000
25%         5.000000
50%         6.000000
75%         6.000000
max         8.000000
Name: quality, dtype: float64
```

In [16]:
```
df.quality.unique()
```

Out[16]:
```
array([5, 6, 7, 4, 8, 3], dtype=int64)
```

In [15]:
```
# Finding NUll Values
null_values = len(df[pd.isna(df['quality'])==True])
print("Null Values: ",null_values)

median = df['quality'].median()
print("Median: ",median)

mode=df['quality'].mode().values[0]
print("Mode: ",mode)
```

```
Null Values:  0
Median:  6.0
```

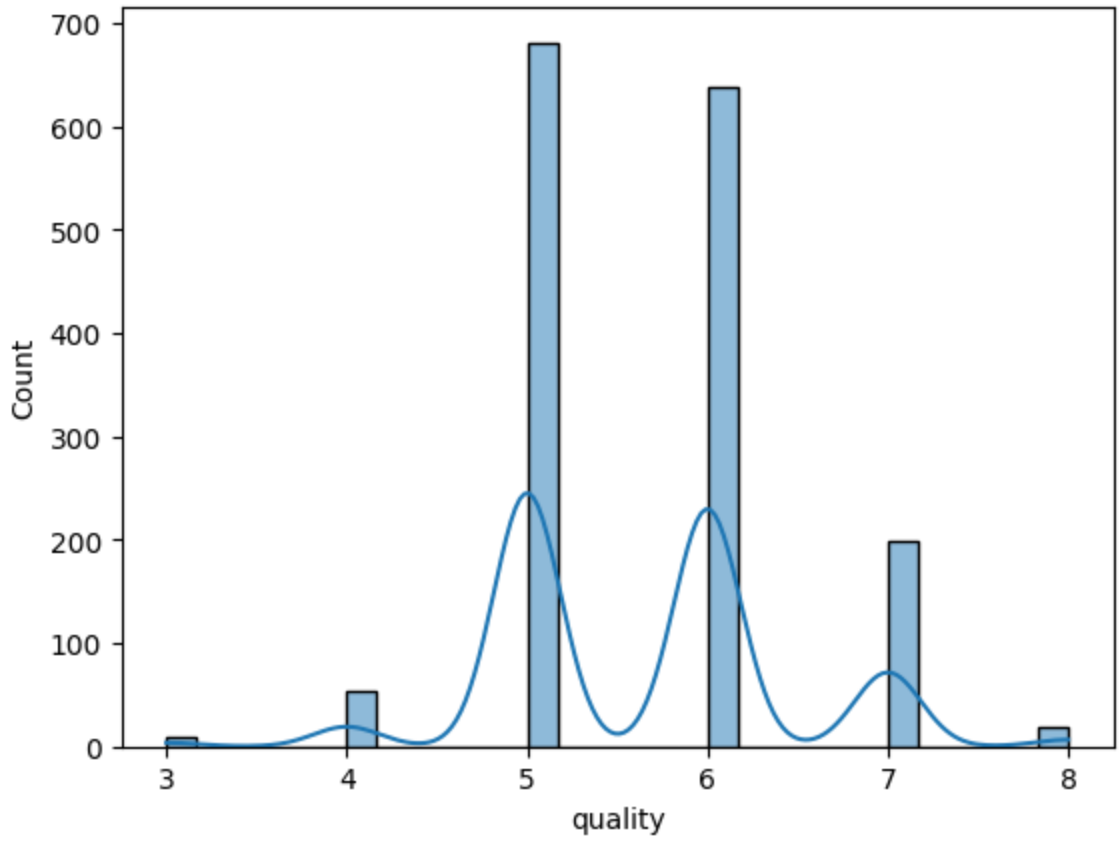
```
Mode:  5
```

We can classify Quality as a target variable and also see that it has discrete values.

Visualizing the data for variable

In [84]:
```
sns.histplot(data = df,x = 'quality',kde = True)
```

Out[84]:
```
<Axes: xlabel='quality', ylabel='Count'>
```



The above plot indicates the mode for data is 5. Hence we can infer that a quality score of 5 was the most repeated followed by a score of 6