# Wine Quality EDA

## Table of Contents :



# Getting started

## Importing libraries

In [ ]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
plt.style.available
plt.style.use('seaborn-pastel')
import warnings
warnings.filterwarnings("ignore")
```

```
C:\Users\Thanh\AppData\Local\Temp\ipykernel_77288\1651659601.py:6: MatplotlibDeprecationWarning: The seaborn styles shipped by Matplotlib are deprecated since 3.6, as they no longer correspond to the styles shipped by seaborn. However, they will remain available as 'seaborn-v0_8-<style>'. Alternatively, directly use the seaborn API instead.
  plt.style.use('seaborn-pastel')
```

## Importing dataset

In [ ]:
```
df = pd.read_csv('wineqt.csv', index_col='Id')
```

## Quick overview about data

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

Out[ ]:

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

No categoricial data within the dataset

## Check Null

In [ ]:
```
null_counts = df.isnull().sum()

print(null_counts)
```

```
fixed acidity           0
volatile acidity        0
citric acid             0
residual sugar          0
chlorides               0
free sulfur dioxide     0
total sulfur dioxide    0
density                 0
pH                      0
sulphates               0
alcohol                 0
quality                 0
dtype: int64
```

## Check for missing data and data types

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

```
<class 'pandas.core.frame.DataFrame'>
Index: 1143 entries, 0 to 1597
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   fixed acidity         1143 non-null   float64
 1   volatile acidity      1143 non-null   float64
 2   citric acid           1143 non-null   float64
 3   residual sugar        1143 non-null   float64
 4   chlorides             1143 non-null   float64
 5   free sulfur dioxide   1143 non-null   float64
 6   total sulfur dioxide  1143 non-null   float64
 7   density               1143 non-null   float64
 8   pH                    1143 non-null   float64
 9   sulphates             1143 non-null   float64
 10  alcohol               1143 non-null   float64
 11  quality               1143 non-null   int64  
dtypes: float64(11), int64(1)
memory usage: 116.1 KB
```

There are no missing values in the dataset. The target column consists of integer values which have an ordinary character. The rest of the values are float. <\html>

## Check Duplicate

In [ ]:
```
data = df.drop(['quality'], axis=1)
duplicate_count = data.duplicated().sum()
print("Number of duplicate rows:", duplicate_count)
```

```
Number of duplicate rows: 125
```

# Analysis of statistic distribution

Conduct exploratory data analysis (EDA) to gain insights into the dataset

a) Calculate basic statistics for numerical columns:

In [ ]:
```
df.describe()
```

Out[ ]:

| | fixed acidity | volatile acidity | citric acid | residual sugar | chlorides | free sulfur dioxide | tota |
|---|---|---|---|---|---|---|---|
| **count** | 1143.000000 | 1143.000000 | 1143.000000 | 1143.000000 | 1143.000000 | 1143.000000 | 1143 |
| **mean** | 8.311111 | 0.531339 | 0.268364 | 2.532152 | 0.086933 | 15.615486 | 45 |
| **std** | 1.747595 | 0.179633 | 0.196686 | 1.355917 | 0.047267 | 10.250486 | 32 |
| **min** | 4.600000 | 0.120000 | 0.000000 | 0.900000 | 0.012000 | 1.000000 | 6 |
| **25%** | 7.100000 | 0.392500 | 0.090000 | 1.900000 | 0.070000 | 7.000000 | 21 |
| **50%** | 7.900000 | 0.520000 | 0.250000 | 2.200000 | 0.079000 | 13.000000 | 37 |
| **75%** | 9.100000 | 0.640000 | 0.420000 | 2.600000 | 0.090000 | 21.000000 | 61 |
| **max** | 15.900000 | 1.580000 | 1.000000 | 15.500000 | 0.611000 | 68.000000 | 289 |

b) Generate summary statistics for categorical columns

In [ ]:

```
# Calculate the frequency counts for each quality value
quality_counts = df['quality'].value_counts()

# Calculate the percentage distribution of quality values
quality_percentage = (quality_counts / len(df)) * 100

# Display the results
print("Frequency counts of quality values:")
print(quality_counts)
print("\nPercentage distribution of quality values:")
print(quality_percentage)
```

```
Frequency counts of quality values:
quality
5    483
6    462
7    143
4     33
8     16
3      6
Name: count, dtype: int64

Percentage distribution of quality values:
quality
5    42.257218
6    40.419948
7    12.510936
4     2.887139
8     1.399825
3     0.524934
Name: count, dtype: float64
```

c) Explore the relationships between variables using appropriate EDA techniques such as correlation analysis or cross-tabulation.

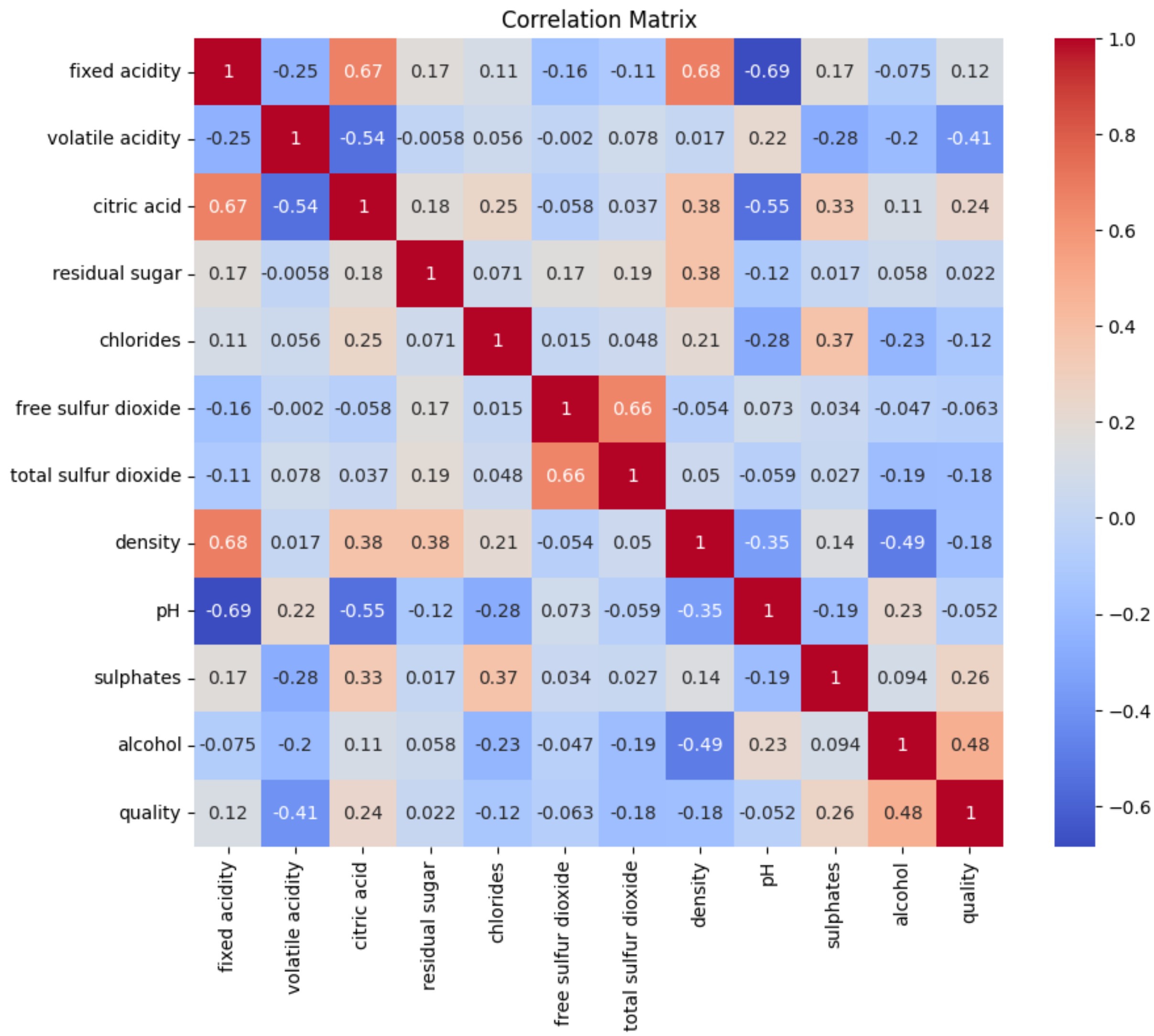
In [ ]:

```
# Correlation matrix
correlation_matrix = df.corr()
print(correlation_matrix)
```

```
                      fixed acidity  volatile acidity  citric acid  
fixed acidity              1.000000         -0.250728     0.673157  \
volatile acidity          -0.250728          1.000000    -0.544187   
citric acid                0.673157         -0.544187     1.000000   
residual sugar             0.171831         -0.005751     0.175815   
chlorides                  0.107889          0.056336     0.245312   
free sulfur dioxide       -0.164831         -0.001962    -0.057589   
total sulfur dioxide      -0.110628          0.077748     0.036871   
density                    0.681501          0.016512     0.375243   
pH                        -0.685163          0.221492    -0.546339   
sulphates                  0.174592         -0.276079     0.331232   
alcohol                   -0.075055         -0.203909     0.106250   
quality                    0.121970         -0.407394     0.240821   

                      residual sugar  chlorides  free sulfur dioxide  
fixed acidity               0.171831   0.107889            -0.164831  \
volatile acidity           -0.005751   0.056336            -0.001962   
citric acid                 0.175815   0.245312            -0.057589   
residual sugar              1.000000   0.070863             0.165339   
chlorides                   0.070863   1.000000             0.015280   
free sulfur dioxide         0.165339   0.015280             1.000000   
total sulfur dioxide        0.190790   0.048163             0.661093   
density                     0.380147   0.208901            -0.054150   
pH                         -0.116959  -0.277759             0.072804   
sulphates                   0.017475   0.374784             0.034445   
alcohol                     0.058421  -0.229917            -0.047095   
quality                     0.022002  -0.124085            -0.063260   

                      total sulfur dioxide   density        pH  sulphates  
fixed acidity                    -0.110628  0.681501 -0.685163   0.174592  \
volatile acidity                  0.077748  0.016512  0.221492  -0.276079   
citric acid                       0.036871  0.375243 -0.546339   0.331232   
residual sugar                    0.190790  0.380147 -0.116959   0.017475   
chlorides                         0.048163  0.208901 -0.277759   0.374784   
free sulfur dioxide               0.661093 -0.054150  0.072804   0.034445   
total sulfur dioxide              1.000000  0.050175 -0.059126   0.026894   
density                           0.050175  1.000000 -0.352775   0.143139   
pH                               -0.059126 -0.352775  1.000000  -0.185499   
sulphates                         0.026894  0.143139 -0.185499   1.000000   
alcohol                          -0.188165 -0.494727  0.225322   0.094421   
quality                          -0.183339 -0.175208 -0.052453   0.257710   

                       alcohol   quality  
fixed acidity        -0.075055  0.121970  
volatile acidity     -0.203909 -0.407394  
citric acid           0.106250  0.240821  
residual sugar        0.058421  0.022002  
chlorides            -0.229917 -0.124085  
free sulfur dioxide  -0.047095 -0.063260  
total sulfur dioxide -0.188165 -0.183339  
density              -0.494727 -0.175208  
pH                    0.225322 -0.052453  
sulphates             0.094421  0.257710  
alcohol               1.000000  0.484866  
quality               0.484866  1.000000  
```

### Heatmap of correlation matrix

In [ ]:
```python
correlation = df.corr()

# Generate a heatmap using seaborn
plt.figure(figsize=(10, 8))
sns.heatmap(correlation, annot=True, cmap='coolwarm', square=True)
plt.title('Correlation Matrix')
plt.show()
```




Some colinearities can be found. PCA could be usefull when creating a machine learning model.

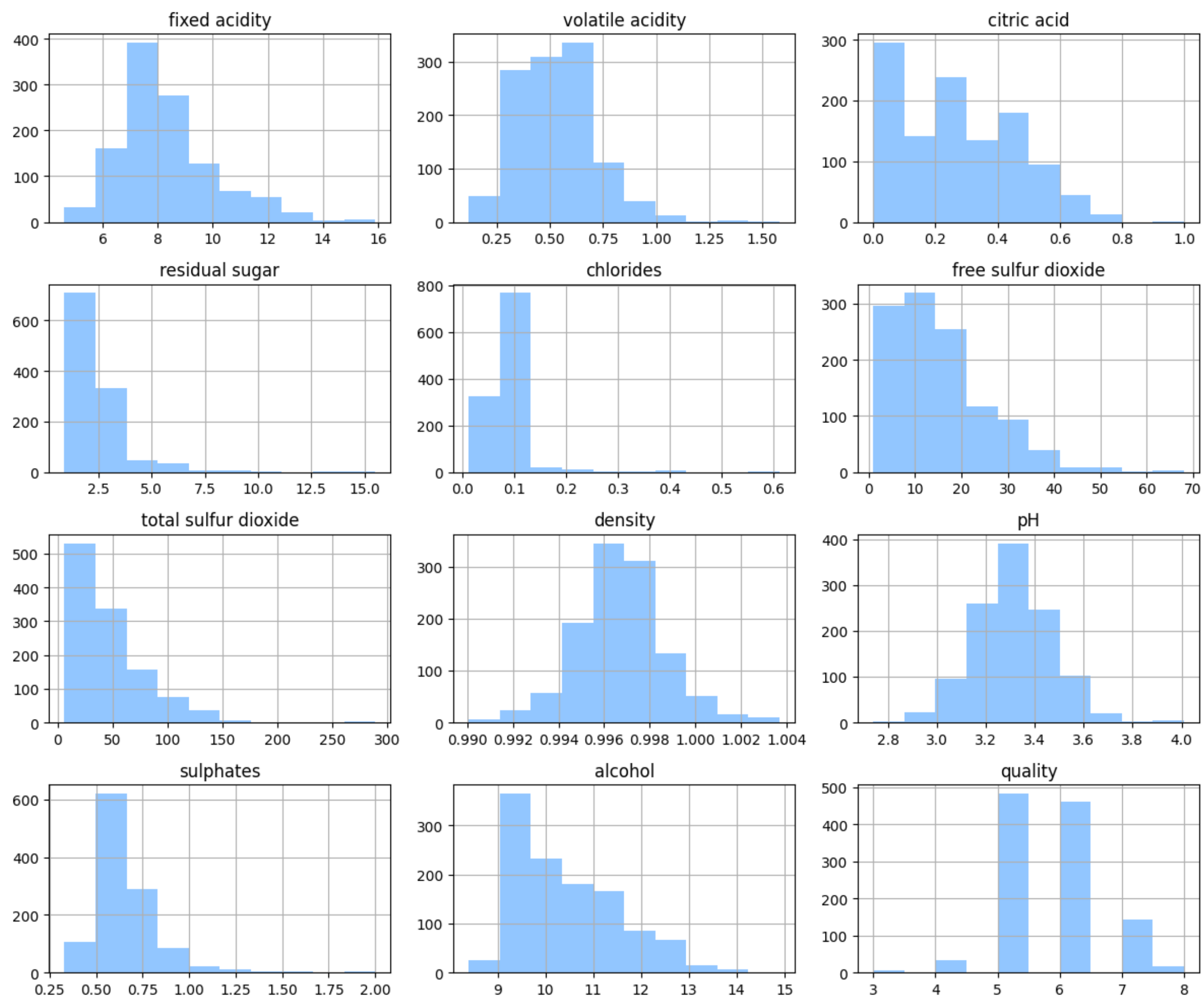
The quality range ranges between 3 and 8. No unrealistic values detectable. <\html>

# Visualisation

# Create meaningful visualizations to understand the dataset better. Include at least three different types of visualization

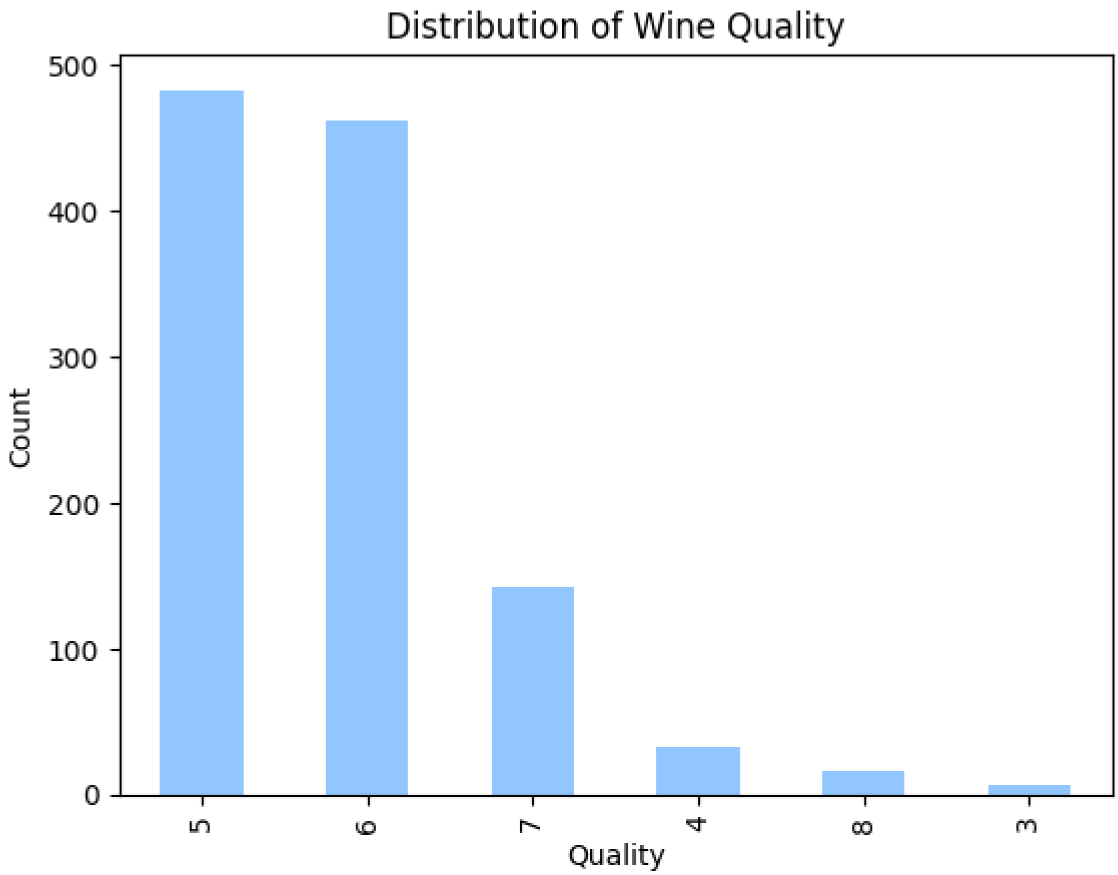
a) A histogram to visualize the distribution of a numerical variable.

In [ ]:
```python
# Histograms
df.hist(figsize=(12, 10))
plt.tight_layout()
plt.show()
```



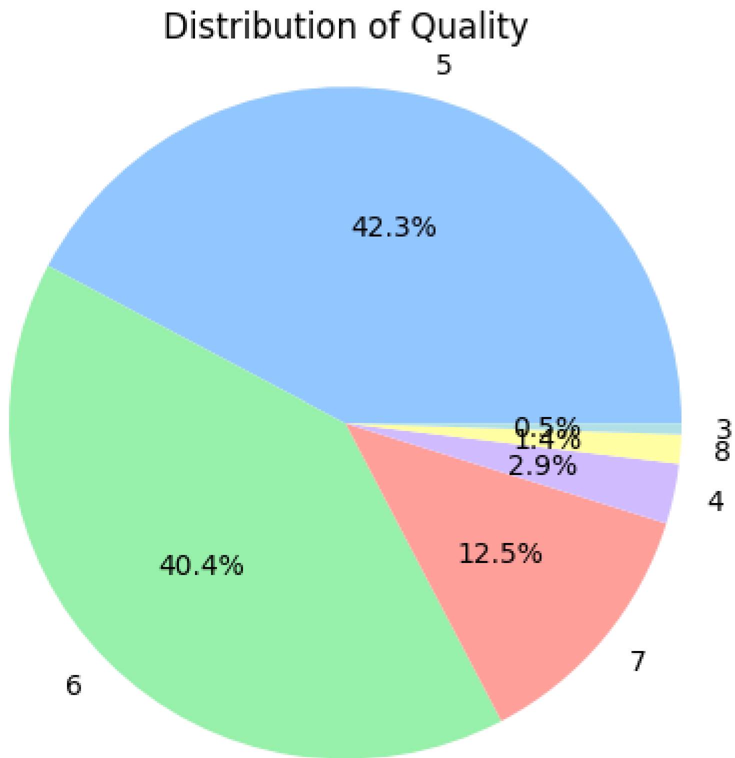

Bar plot for 'quality' column

In [ ]:
```python
df['quality'].value_counts().plot(kind='bar')
plt.xlabel('Quality')
plt.ylabel('Count')
plt.title('Distribution of Wine Quality')
plt.show()
```




Distribution of Quality

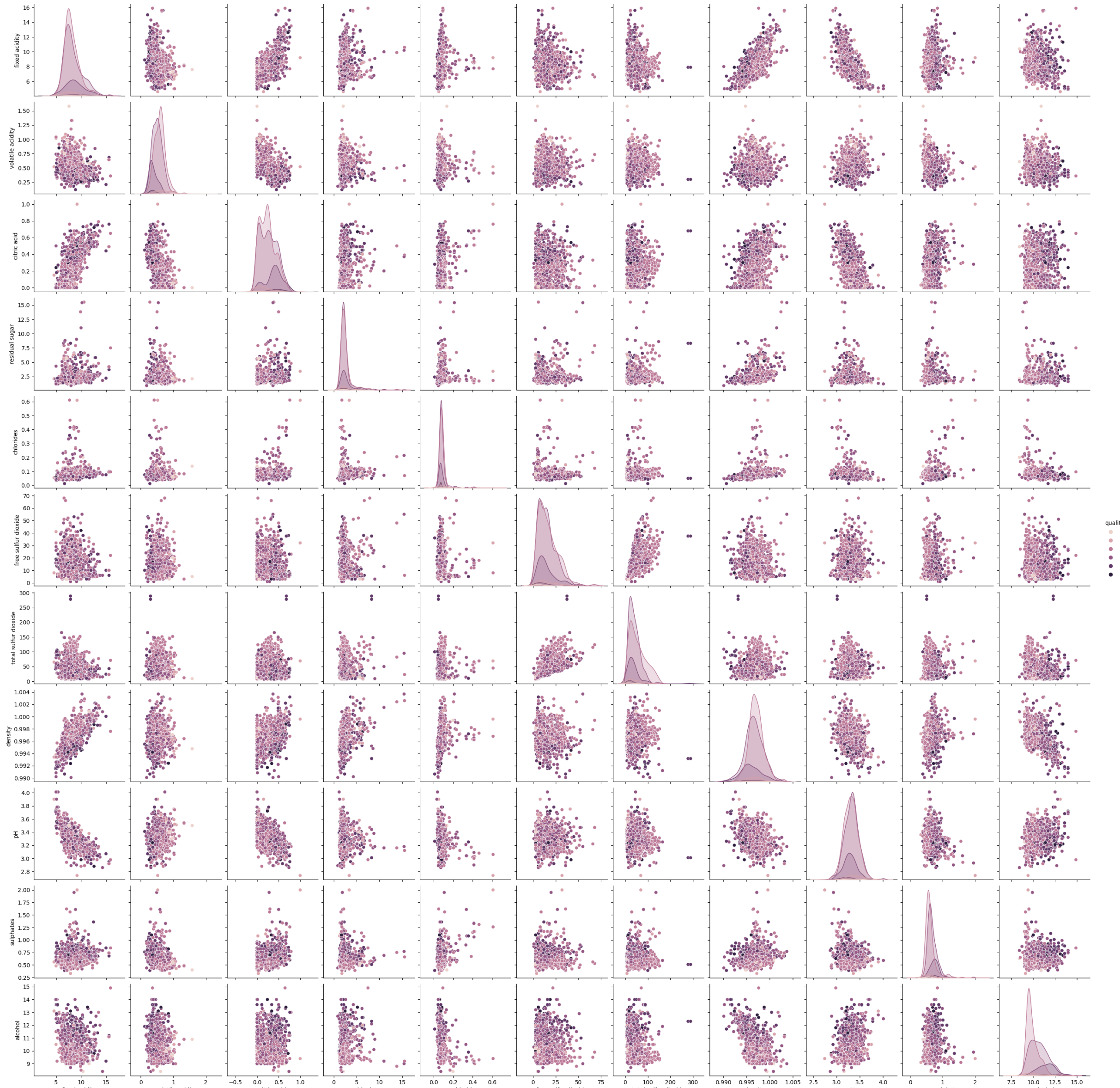
In [ ]:
```
value_counts = df['quality'].value_counts()

# Plot the distribution of the "quality" column as a pie chart
plt.pie(value_counts, labels=value_counts.index, autopct='%1.1f%%')
plt.title('Distribution of Quality')
plt.axis('equal')  # Ensure pie is drawn as a circle
plt.show()
```




Pairplot

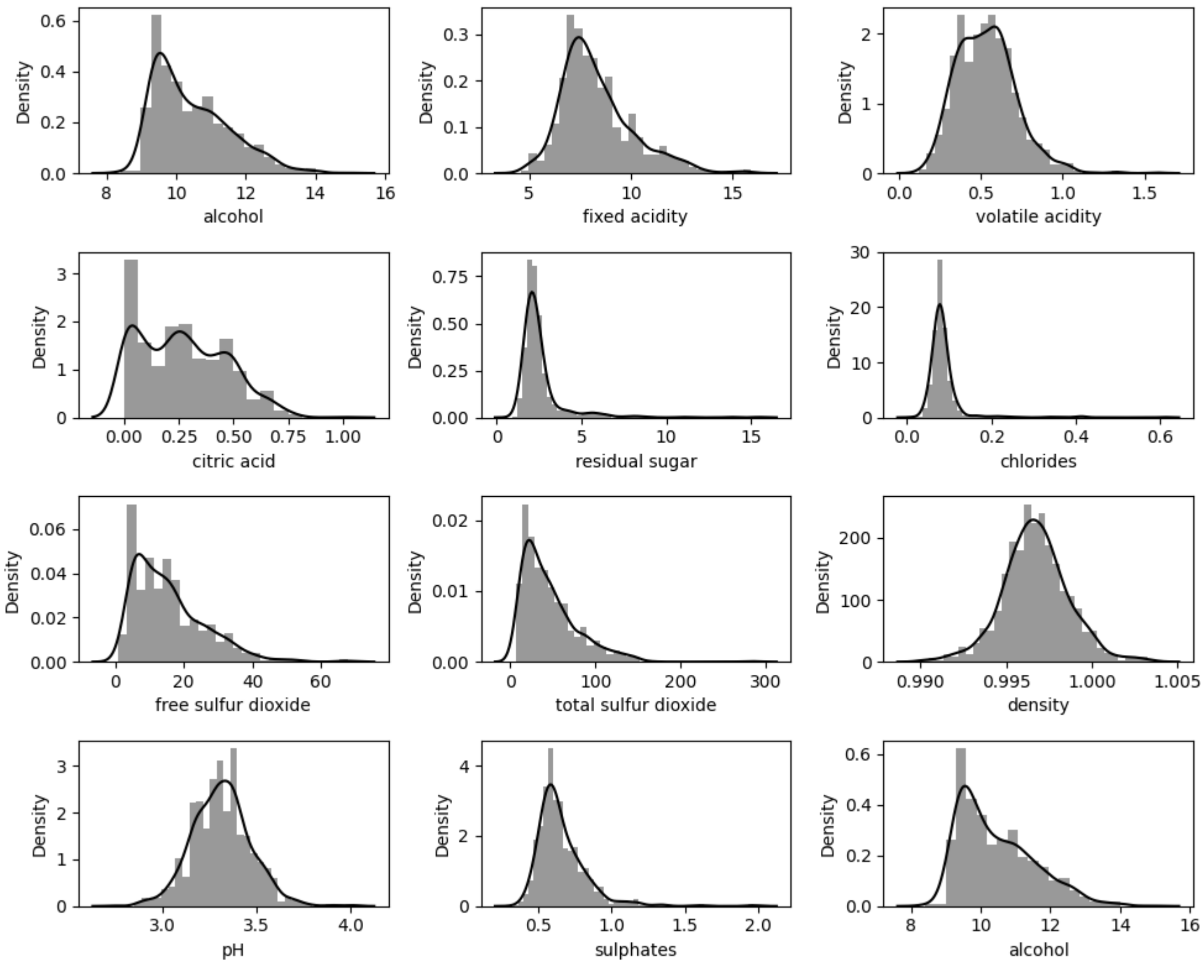
In [ ]:
```
sns.pairplot(df,hue='quality')
```

Out[ ]:
```
<seaborn.axisgrid.PairGrid at 0x2d5cf7f7c90>
```



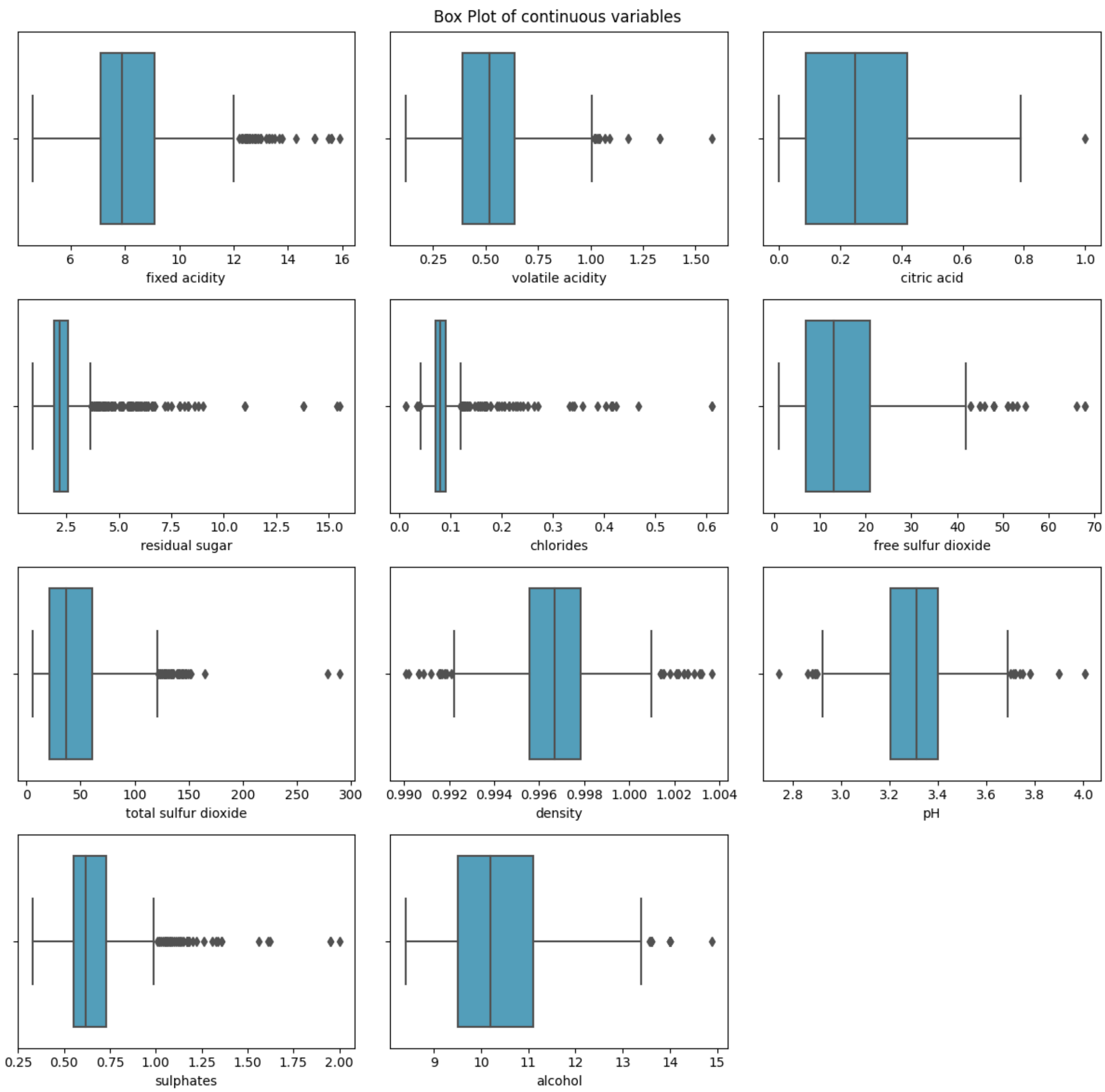

Some correlations can be observed, which can be looked at in more detail in a heatmap. For quality column (target), no strong linear trend can be observed for single attributes.

In [ ]:
```
fig, axes = plt.subplots(4,3,figsize=(10,8))

i=-1
for row in range(4):
    for col in range(3):
        if i<11:
            sns.distplot(df[df.columns[:-1][i]], ax=axes[row][col], color='black')
            i+=1

fig.tight_layout()
```




Box plot for each continuous variable

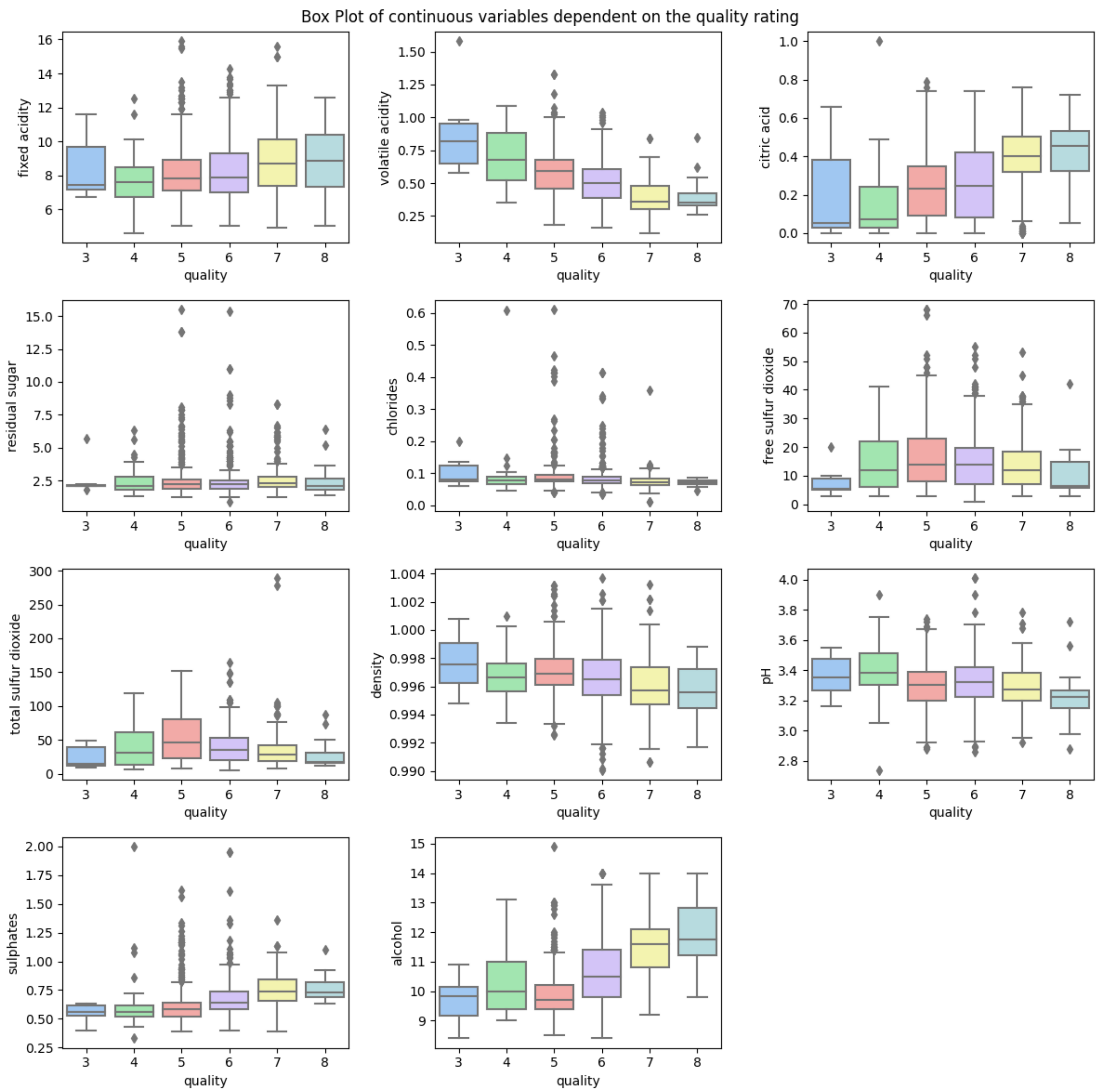
In [ ]:
```
plt.figure(figsize=(12,12))
for i, col in enumerate(df.select_dtypes(include=['float64']).columns):
    ax = plt.subplot(4,3, i+1)
    sns.boxplot(data=df, x=col, ax=ax,palette="GnBu_d")
plt.suptitle('Box Plot of continuous variables')
plt.tight_layout()
```




Several outliers can be detected in the dataset using boxplots. However, values don't seem to be unusual. For machine learning models, I would keep them in the first place.

Box plot for each continuous variable in dependance of target column

In [ ]:
```
plt.figure(figsize=(12,12))
for i, col in enumerate(df.select_dtypes(include=['float64']).columns):
    ax = plt.subplot(4,3, i+1)
    sns.boxplot(data=df, x='quality', y=col, ax=ax)
plt.suptitle('Box Plot of continuous variables dependent on the quality rating')
plt.tight_layout()
```




We get several insights of this figure:

1. Median of fixed acidity increases with increasing quality rating, whereas volatile acidity decreases.
2. median of the amount of citric acid increases with increasing quality rating.
3. residual sugar, chlorides and density seem to have little effect on quality rating.
4. low and high rated wines seem to be low in free sulfur dioxide and total sulfur dioxide.
5. better rated wines seem to have a lower pH.
6. wines with higher ratings seem to be higher in the amount of sulphates and alcohol.

# Conclusion

**Report - Wine Data**

**Introduction:** This report focuses on the analysis of a dataset related to wine, including attributes such as fixed acidity, volatile acidity, citric acid, residual sugar, chloride, free sulfur dioxide, density, pH, sulphates, alcohol, quality, and several other attributes. Below is a summary of the findings after data wrangling, exploratory data analysis (EDA), and visualizations.

**1. Data Inspection and Preprocessing:**

- The raw data appeared to be clean, with no significant missing values or obvious errors in the given columns.
- During preprocessing, there was little need for data cleaning.

**2. Data Distribution:**

- The dataset contains both continuous and categorical attributes, with 'quality' as the target variable. 'Quality' is a categorical variable representing the quality of wine on a scale from 3 to 9.
- The distribution of wine quality shows signs of skewness, with a high concentration of wines having quality around 5 and 6.

**3. Data Visualization:**

- Visualizations such as frequency plots, box plots, and scatter plots were created to explore the distribution and relationships between attributes and wine quality.
- Notable findings include a potential relationship between alcohol content and wine quality, as well as the influence of certain chemical attributes (e.g., volatile acidity, citric acid, and sulphates) on wine quality.

**4. Feature Engineering Techniques:**

- Additional feature engineering may be required to better understand the relationships between attributes and wine quality. For example, creating new features or normalizing data for modeling purposes.

**5. Modeling:**

- After completing data preprocessing and exploratory data analysis, the next steps typically involve building predictive models to understand the key factors influencing wine quality.

**6. Conclusion:**

This project has provided a foundation for in-depth analysis and modeling, allowing you to identify the key factors that influence wine quality and make predictions based on this information. Subsequent steps may include building predictive models to provide quality predictions based on chemical and mechanical attributes.
This dataset shows the rating of >1000 different wines and their chemical parameters. The dataset is unbalanced regarding the different quality ratings. >80 % of the wines get a rating of "5" or "6" which can be translated as average rated wines.

There are no missing values within the dataset and no categorical columns.
Colinearity between different features can be observed. For machine learning modells, a PCA could be helpfull.
There are some outliers within the values. However, the statistical analysis does not show unrealistic values, so for machine learning, I would prefer to not delete or manipulate them in the first place.
When we look at the different features and their impact on the wine rating, different trends, as well as apparently little effect can be observed(see chapter before). When building a machine learning model, alcohol and volatile acidity will probably have the highest feature importance as the trend can be clearly seen.