Assignment 4

Name – Sachin

Reg no – 21BAC10036

Campus – Vit Bhopal

Code:

1. Load the Dataset

import pandas as pd

dataset = pd.read_csv('winequality-red.csv')

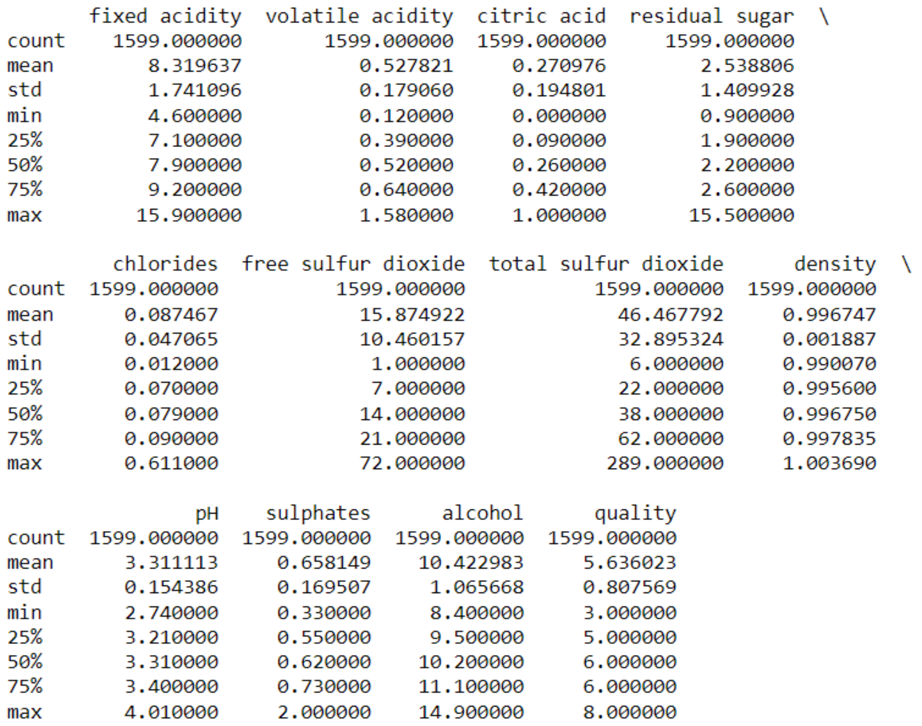
2. Data preprocessing including visualization

# Display summary statistics of the dataset

summary_stats = dataset.describe()

print(summary_stats)

Output:



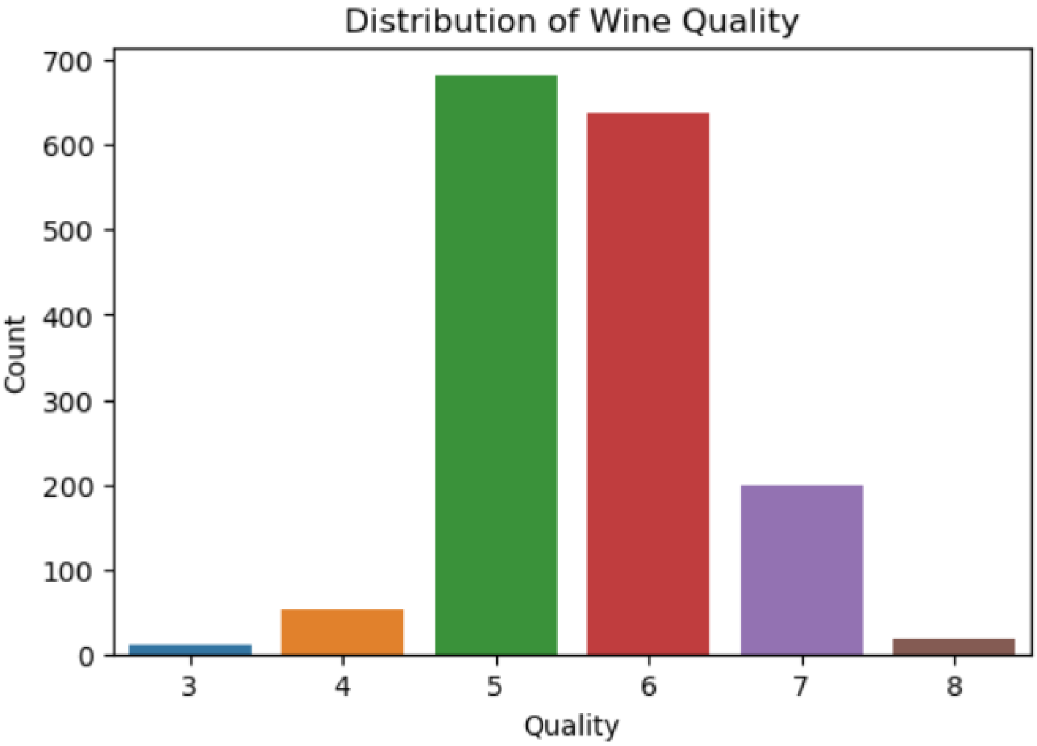
```
       fixed acidity  volatile acidity  citric acid  residual sugar  \
count    1599.000000       1599.000000  1599.000000     1599.000000   
mean        8.319637          0.527821     0.270976        2.538806   
std         1.741096          0.179060     0.194801        1.409928   
min         4.600000          0.120000     0.000000        0.900000   
25%         7.100000          0.390000     0.090000        1.900000   
50%         7.900000          0.520000     0.260000        2.200000   
75%         9.200000          0.640000     0.420000        2.600000   
max        15.900000          1.580000     1.000000       15.500000   

         chlorides  free sulfur dioxide  total sulfur dioxide      density  \
count  1599.000000          1599.000000           1599.000000  1599.000000   
mean      0.087467            15.874922             46.467792     0.996747   
std       0.047065            10.460157             32.895324     0.001887   
min       0.012000             1.000000              6.000000     0.990070   
25%       0.070000             7.000000             22.000000     0.995600   
50%       0.079000            14.000000             38.000000     0.996750   
75%       0.090000            21.000000             62.000000     0.997835   
max       0.611000            72.000000            289.000000     1.003690   

                pH    sulphates      alcohol      quality  
count  1599.000000  1599.000000  1599.000000  1599.000000  
mean      3.311113     0.658149    10.422983     5.636023  
std       0.154386     0.169507     1.065668     0.807569  
min       2.740000     0.330000     8.400000     3.000000  
25%       3.210000     0.550000     9.500000     5.000000  
50%       3.310000     0.620000    10.200000     6.000000  
75%       3.400000     0.730000    11.100000     6.000000  
max       4.010000     2.000000    14.900000     8.000000  
```

import matplotlib.pyplot as plt

import seaborn as sns

```
# Countplot of wine quality
plt.figure(figsize=(6, 4))
sns.countplot(data=dataset, x='quality')
plt.title('Distribution of Wine Quality')
plt.xlabel('Quality')
plt.ylabel('Count')
plt.show()
```

Output:

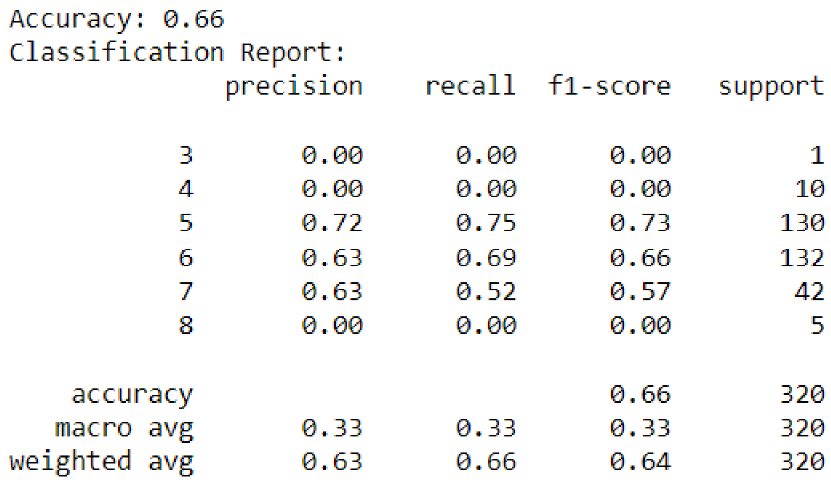


3 & 4. Machine Learning Model building and Evaluate the model

```
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report

X = dataset.drop('quality', axis=1)
y = dataset['quality']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train, y_train)

y_pred = clf.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)

print(f'Accuracy: {accuracy:.2f}')
print('Classification Report:')
print(classification_rep)
```

Output:

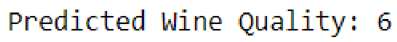


```
Accuracy: 0.66
Classification Report:
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.00      0.00      0.00        10
           5       0.72      0.75      0.73       130
           6       0.63      0.69      0.66       132
           7       0.63      0.52      0.57        42
           8       0.00      0.00      0.00         5

    accuracy                           0.66       320
   macro avg       0.33      0.33      0.33       320
weighted avg       0.63      0.66      0.64       320
```

5. # Create a random observation

```
new_observation = pd.DataFrame({
    'fixed acidity': [7.0],
    'volatile acidity': [0.4],
    'citric acid': [0.25],
    'residual sugar': [2.0],
    'chlorides': [0.045],
    'free sulfur dioxide': [35.0],
    'total sulfur dioxide': [120.0],
    'density': [0.99],
    'pH': [3.2],
    'sulphates': [0.6],
    'alcohol': [11.0]
})

# Use the trained model to predict the quality of the new observation
predicted_quality = clf.predict(new_observation)
print(f'Predicted Wine Quality: {predicted_quality[0]}')
```

Output:



```
Predicted Wine Quality: 6
```