SRI LANKA INSTITUTE OF INFORMATION TECHNOLOGY

MACHINE LEARNING

(SE4060)

Assignment – Report

Name: Liyanage L.C.

Registration Number: IT15025722

Batch: Software Engineering(WE)

Problem Addressed

Predicting the quality of red wine using the features of the wine identified through physicochemical tests.

Description of the dataset

Link to the dataset: http://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/

The dataset was taken from UCI machine learning repository. They have provided two sets as red wine quality and white wine quality. I have used red wine quality dataset in this assignment.

The red wine quality dataset is related to red variant of the Portuguese "Vinho Verde" wine. It consists of 1599 red wine instances. There are 11 attributes and 1 output attribute. The input variables are,

1.Fixed acidity	2.Volatile acidity	3.Citric acid	4.Residual sugar	5.Chlorides
6.Free Sulphur dioxide	7.Total sulfur dioxide	8. Density	9.pH	10.Sulphates
11.Alcohol				

The input variables have been derived based on physicochemical tests. The output variable is quality (score between 0 and 10) which has been derived based on sensory data. Each expert graded the wine quality between 0 (very bad) and 10 (very excellent).

The red wine quality dataset has no missing attribute values.

Data Preprocessing

Since the red wine quality dataset doesn't contain missing attribute values any preprocessing, in order to remove them was not carried out.

The dataset was tested for null values and correct data types. Feature scaling was carried out in order to normalize the data. Here the features are fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates and alcohol. But it was seen that the accuracy of the classification without feature scaling was higher than the accuracy of the classification with feature scaling. Therefore, it can be concluded that data scaling did not improve the performance of the used classification and I have used the unscaled version of the dataset in order to construct the model.

Classification using Decision Trees

Decision tree is a predictive model used in machine learning, statistics and data mining. A Decision tree consists of a tree structure (Directed, acyclic graph) starting with some observations about an item (Data) to conclusions (target value).

Decision Trees are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

The red wine quality dataset can be used for bother classification and regression. I have used the dataset for classification to complete the assignment. With respect to the wine quality dataset the target variable is quality and the data features are fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates and alcohol. Therefore, decision tree approach has been used in order to predict the quality of red wine from the data features mentioned above.

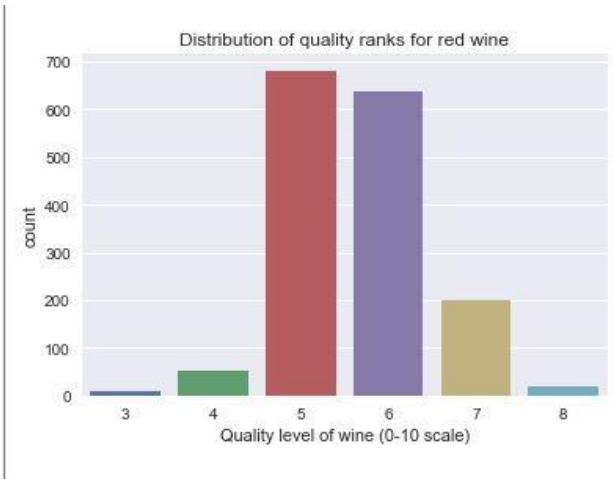
I have used DecisionTreeClassifier in order to predict the quality of red wine using the provided data features. DecisionTreeClassifier is a class capable of performing multi-class classification on a dataset. Since there are 11 data features which affect the quality of the wine, those can be considered as the multiple classes that are being considered in the classification that I have carried out.

Results

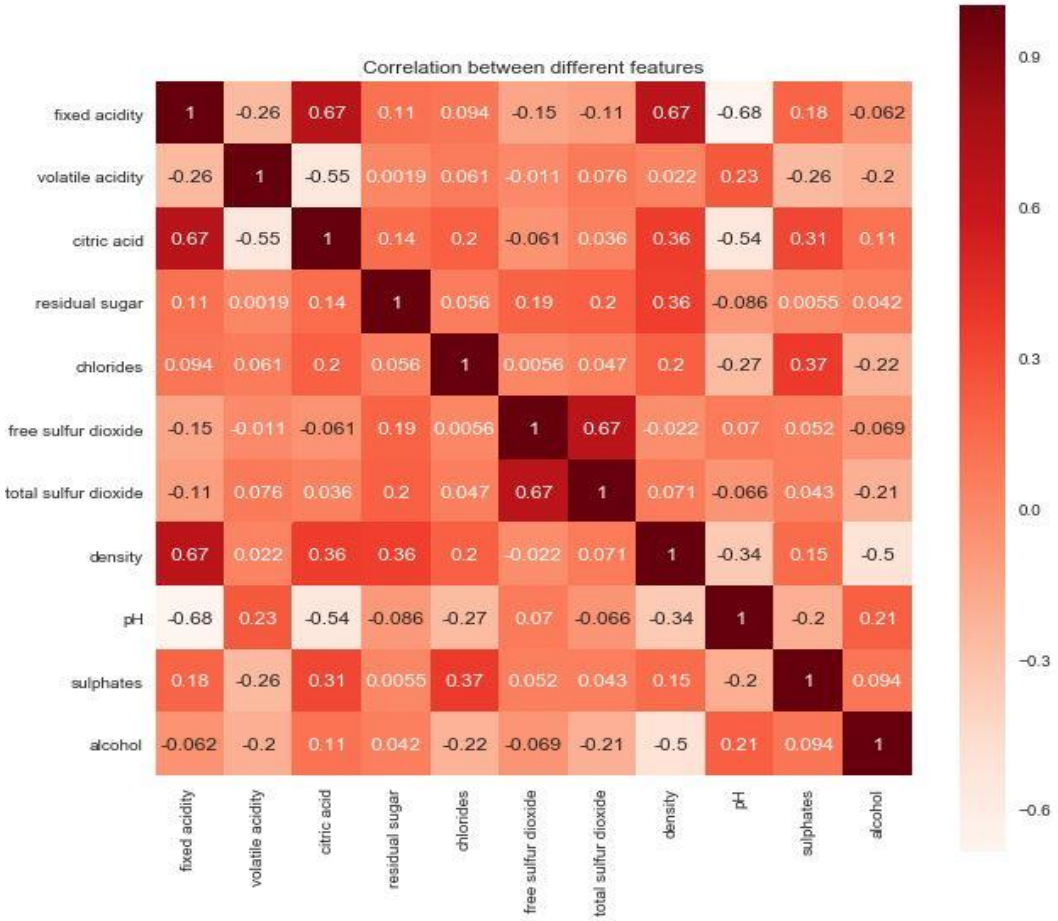
The dataset was tested for null values and checked for correct datatypes as a part of preprocessing.

```
Test for null values and check correct datatypes
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1599 entries, 0 to 1598
Data columns (total 12 columns):
fixed acidity           1599 non-null float64
volatile acidity        1599 non-null float64
citric acid             1599 non-null float64
residual sugar          1599 non-null float64
chlorides               1599 non-null float64
free sulfur dioxide     1599 non-null float64
total sulfur dioxide    1599 non-null float64
density                 1599 non-null float64
pH                      1599 non-null float64
sulphates               1599 non-null float64
alcohol                 1599 non-null float64
quality                 1599 non-null int64
dtypes: float64(11), int64(1)
memory usage: 150.0 KB
```

In order to visualize the distribution of quality ranks of red wine a bar plot was plotted. According to the below figure it can be inferred that the quality rating of the red wine follows a fairly normal distribution.

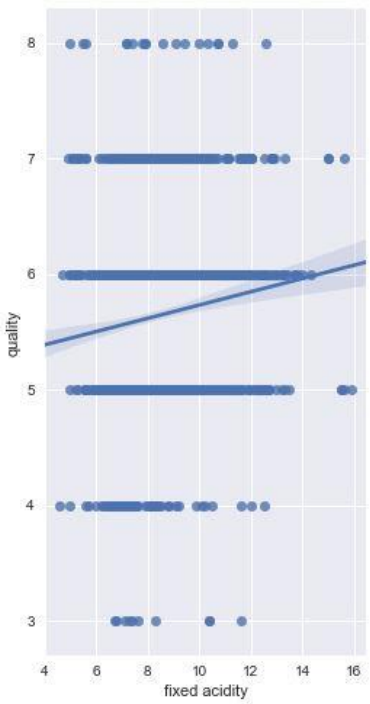


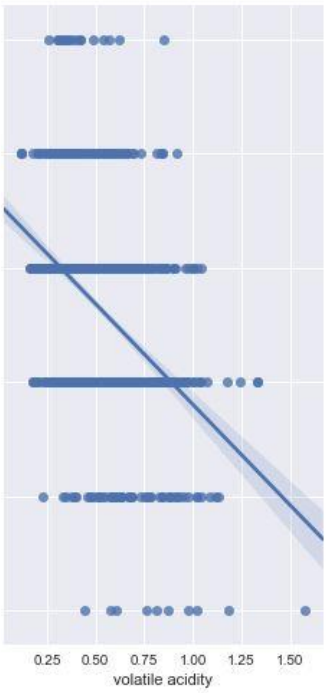
In order to understand further connections of the dataset a heatmap consisting of the correlation coefficients between features was plotted.

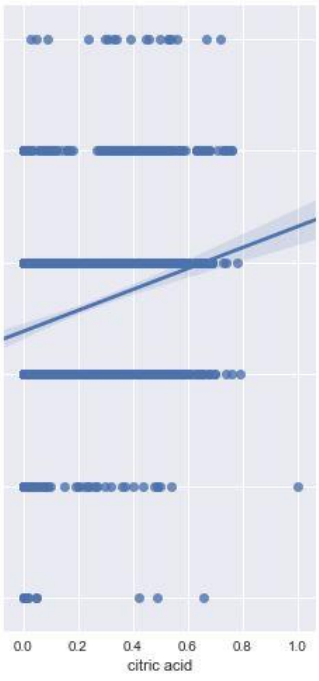


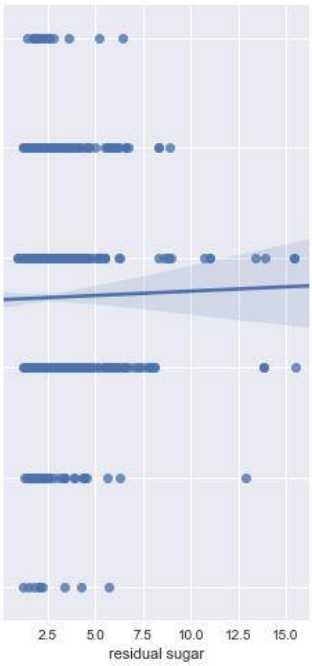
When we look at the above heatmap it can be identified that there are strong positive correlations between fixed acidity and density and total Sulphur dioxide and free Sulphur dioxide. And also, there strong negative correlation between fixed density and pH, citric acid and pH.

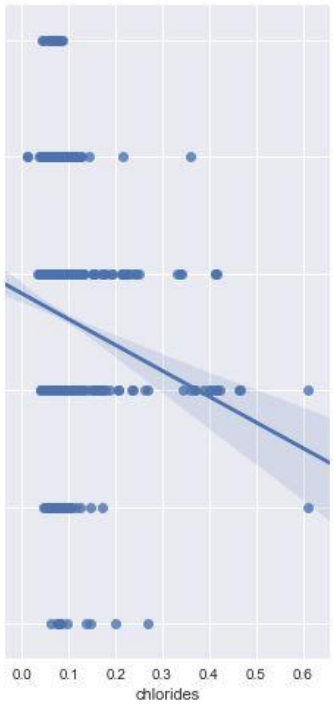
The below given diagrams depicts the regression between different features and targets.

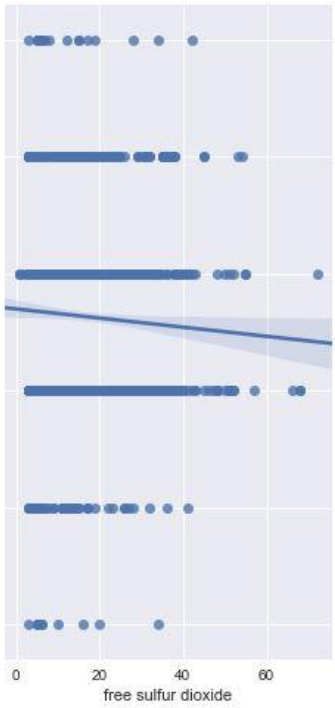


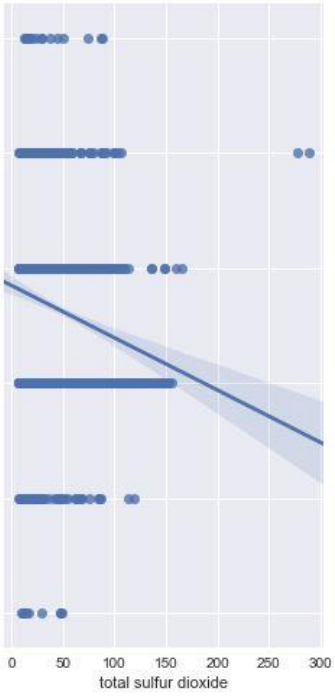


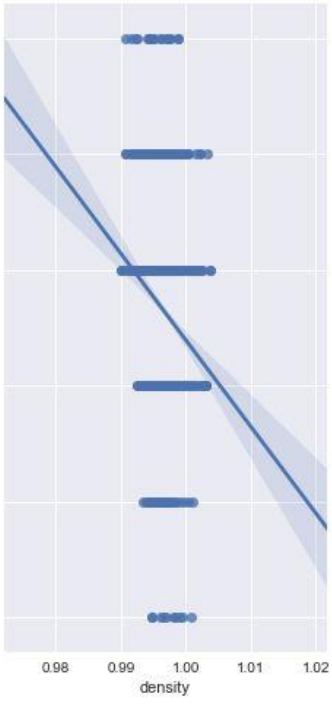




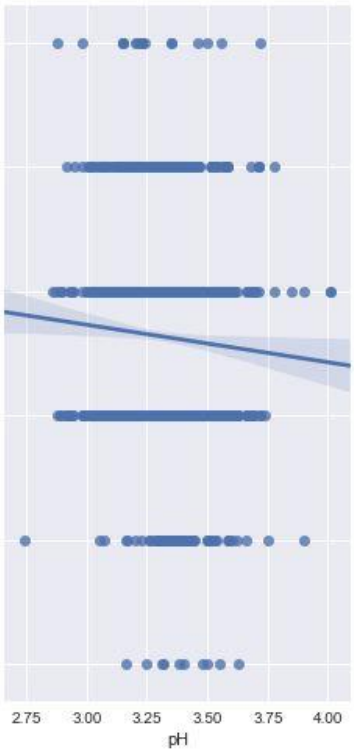

0.0
0.1
0.2
0.3
0.4
0.5
0.6
chlorides

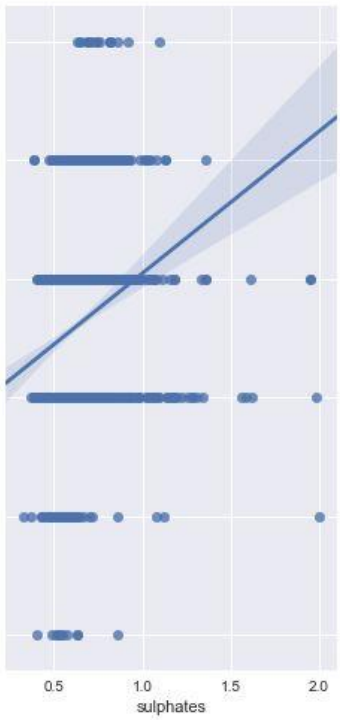

0
20
40
60
free sulfur dioxide


0
50
100
150
200
250
300
total sulfur dioxide

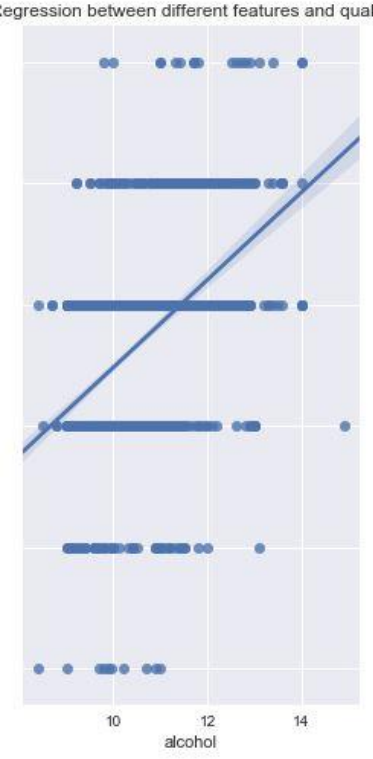

0.98
0.99
1.00
1.01
1.02
density

R





egression between different features and qualit



The predicted quality of red wine using the DecisionTreeClassifier is shown below. For the easiness of reading only 20 predictions and their expected or the true values have been displayed.

```
Predicted quality of first twenty wines:
6
5
7
5
5
6
5
6
5
5
5
5
6
6
6
7
7
5
6
6
```

```
Expected quality first twenty wines:
1109    6
1032    5
1002    7
487     6
979     5
1054    6
542     5
853     6
1189    4
412     5
1099    5
475     5
799     6
553     5
1537    6
1586    6
805     7
1095    5
1547    5
18      4
Name: quality, dtype: category
Categories (6, int64): [3, 4, 5, 6, 7, 8]
```

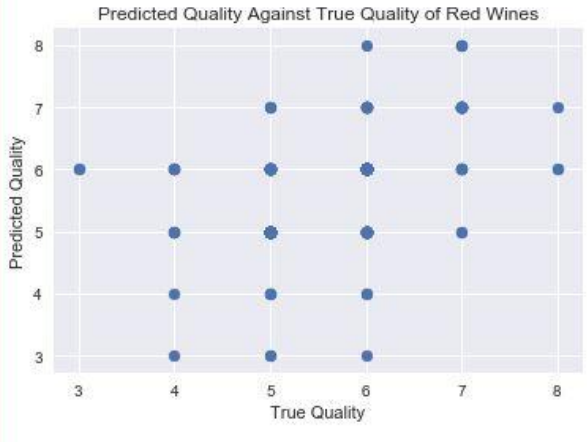
When we analyze the predicted values and the expected values it can be seen that many of them are predicted correctly while some are not.

Therefore, the accuracy of the DecisionTreeClassifier needs to be taken into consideration.

```
Mean accuracy on the given test data and label: 0.659375

Mean accuracy on the given test data and label(As a percentage): 66%
```

Mean accuracy was calculated as 66% percent which is a considerably acceptable amount.



The above graph shows the predicted quality against true quality for the test dataset.

The decision tree score for the test set was calculated as 68%.

```
Decision Tree score for test set: 0.684375
```

When the classification report was generated in order to show the main classification metrics the below results were obtained.

```
Classification Report
             precision    recall  f1-score   support

          3       0.00      0.00      0.00         2
          4       0.17      0.09      0.12        11
          5       0.72      0.73      0.72       135
          6       0.69      0.67      0.68       142
          7       0.55      0.63      0.59        27
          8       0.00      0.00      0.00         3

avg / total       0.66      0.66      0.66       320
```

The accuracy of the classification was identified through confusion matrix.

```
Confusion Metrix
[[ 0  0  0  2  0  0]
 [ 1  1  4  5  0  0]
 [ 2  3 98 29  3  0]
 [ 1  2 33 95 10  1]
 [ 0  0  2  5 17  3]
 [ 0  0  0  2  1  0]]
```

The mean absolute error and accuracy classification score were identified as 41% and 66% respectively.

```
Mean Absolute Error:
0.4125
```

```
Accuracy Classification Score:
0.659375
```

In order to identify the important features in the classification feature importance finding was carried out and the following results were generated.

```
Feature Importances for Decision tree

fixed acidity: 0.06034534171908537
volatile acidity: 0.10219766700528324
citric acid: 0.07228372696542712
residual sugar: 0.07977069803656936
chlorides: 0.07998464100585306
free sulfur dioxide: 0.08567506516945853
total sulfur dioxide: 0.08610263785043883
density: 0.07976926692590815
pH: 0.07250862275124714
sulphates: 0.11329142302614767
alcohol: 0.16807090954458154
```

Critical Analysis and Discussion

Decision Tree model showed an accuracy of 68% when predicting the quality from the provided features. It is an acceptable level of accuracy and it can be concluded that decision tree model is a considerably good supervised learning model in order to predict quality of red wine from the given dataset. An also, it can be seen that only one feature can't be used to decide upon the quality of the wine.

According to the decision tree model, alcohol content, sulphates and volatile acidity are the most important features in deciding whether a wine must be rated high or not.

In order to improve the accuracy of the decision tree model that I have used, the important features can be considered more in predicting the quality of the red wine. Usage of multiple trees or Random Forests would also be a better approach with respect to the given dataset. K-Fold Cross Validation can be carried out to ensure the validity of the test dataset and it would pave way in improving the performance of the model as well. Using logistic regression after usage of decision trees would also be another approach in increasing the accuracy and performance of the existing model.

Future work includes trying out the above-mentioned methods in order to improve the performance and accuracy of the model and trying out predicting the quality of the red wine by using different other supervised learning approaches such as Support Vector Machine, K-Means Cluster Analysis etc.

Appendix

```
import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pylab as plt

from sklearn.model_selection import train_test_split

from sklearn import tree

from sklearn.metrics import
classification_report,accuracy_score,confusion_matrix,mean_absolute_error

#load the dataset using pandas

red_wine_data = pd.read_csv('winequality-red.csv', sep=';')

#---------------- Data Visualization ----------------------------

print(red_wine_data.head())
```

```
#test for null values and check correct datatypes
print('\nTest for null values and check correct datatypes')
assert red_wine_data.notnull().all().all()
red_wine_data.info()
#---------------- Data Analysis ---------------------------------
#distribution of quality ranks for red wine
red_wine_data['quality'] = pd.Categorical(red_wine_data['quality'])
sns.countplot(x="quality", data=red_wine_data)
plt.xlabel("Quality level of wine (0-10 scale)")
plt.title('Distribution of quality ranks for red wine')
plt.show()
#correlation between different features plotted in a heat map
correlation = red_wine_data.corr()
figure = plt.subplots(figsize=(10,10))
sns.heatmap(correlation,vmax=1,square=True,annot=True,cmap='Reds')
plt.title('Correlation between different features')
plt.show()
features = ['fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar',
    'chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'density',
    'pH', 'sulphates', 'alcohol']
#ploting the regression between different features and label(quality)
sns.pairplot(red_wine_data,x_vars=features,y_vars='quality',kind='reg',size=7,aspect=0.5)
plt.title('Regression between different features and quality')
plt.show()
#------------------- Decision Tree Classification -----------------------------------
#drop target variable
x=red_wine_data.drop('quality', axis=1)
```

```
y=red_wine_data.quality

#split dataset into training and testing data

x_train, x_test, y_train, y_test = train_test_split(x, y,test_size=0.2,train_size=0.8,random_state
= 0)

# Show the results of the split

print("\nTraining set has {} samples.".format(x_train.shape[0]))

print("Testing set has {} samples.".format(x_test.shape[0]))

clf=tree.DecisionTreeClassifier()

#build decision tree classifier from the training set

clf.fit(x_train, y_train)

#predict class or regression value for x

y_pred = clf.predict(x_test)

#converting the numpy array to list

x_list=np.array(y_pred).tolist()

#printing first twenty predictions

print("\nPredicted quality of first twenty wines:")

for i in range(0,20):

   print(x_list[i])

 #printing first twenty expectations

print("\nExpected quality first twenty wines:")

print(y_test.head(n=20))

#mean accuracy on the given test data and label(quality).

confidence = clf.score(x_test, y_test)

print("\nMean accuracy on the given test data and label:",confidence)

print("\nMean accuracy on the given test data and label(As a percentage):
{}%".format(int(round(confidence * 100))))

plt.scatter(y_test, y_pred)

plt.xlabel('True Quality')
```

```
plt.ylabel('Predicted Quality')
plt.title('Predicted Quality Against True Quality of Red Wines')
plt.show()
print('\nDecision Tree score for test set: %f' % clf.fit(x_train, y_train).score(x_test, y_test))
#show main classification metrics
print('\nClassification Report')
print(classification_report(y_test, y_pred))
#confusion matrix to evaluate the accuracy of a classification
print('\nConfusion Metrix')
print(confusion_matrix(y_test, y_pred))
#calculate accuracy classification score
print('\nAccuracy Classification Score:')
print(accuracy_score(y_test, y_pred))
#calculate Mean Absolute Error
print('\nMean Absolute Error:')
print(mean_absolute_error(y_test, y_pred))
tree.export_graphviz(clf, out_file='tree.dot')
print('\nFeature Importances for Decision tree\n')
for importance,feature in zip(clf.feature_importances_,['fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar','chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'density', 'pH', 'sulphates', 'alcohol']):
    print('{}: {}'.format(feature,importance))
```