VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JnanaSangama, Belgaum-590014

A Machine Learning Internship Report
On

"RED WINE QUALITY PREDICTION"

Submitted in Partial fulfillment of the Requirements for the VII Semester of the Degree of

Bachelor of Engineering
In
Computer Science & Engineering
By

MEGHANA G(1CE17CS059)

Under the Guidance of
Mrs. Ambika P R
Asst. Professor, Dept. of CSE

CITY ENGINEERING COLLEGE

Doddakallasandra, Kanakapura Road,
Bengaluru-560061

CITY ENGINEERING COLLEGE

Doddakallasandra, Kanakapura Road, Bengaluru-560061

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the Machine Learning Project work entitled **“RED WINE QUALITY PREDICTION”** has been carried out By **MEGHANA G (1CE17CS059),** bonafide student of City Engineering College in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visveshvaraya Technological University, Belgaum during the year **2019-2020**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The Machine Learning Mini Project Report has been approvedas it satisfies the academic requirements in respect of project work prescribed for the said Degree.

Mrs. Ambika P R	**Mr. B Vivekavardhana Reddy**	**Dr. V. S Rama Murthy**
Asst.Prof, Dept.of CSE	Head, Dept. of CSE	Principal

External Viva

Name of the examiners   Signature with date

1.

2.

ABSTRACT

Nowadays people try to lead a luxurious life. They tend to use the things either for show off or for their daily basis. These days the consumption of red wine is very common to all. So it became important to analyze the quality of red wine before its consumption to preserve human health. Hence this research is a step towards the quality prediction of the red wine using its various attributes. Red wine quality and style are highly influenced by the qualitative and quantitative composition of aromatic compounds having various chemical structures and properties and their interaction within different red wine matrices. The understanding of interactions between the wine matrix and volatile compounds and the impact on the overall flavor as well as on typical or specific aromas is getting more and more important for the creation of certain wine styles. Based on the data visualisation of python processing, classical visualization tools such as boxplot, correlation matrix, jointplot and various algorithms for the result.

ACKNOWLEDGEMENT

While presenting this Machine Learning Project on **“Red Wine Quality Prediction”**, I feel that it is my duty to acknowledge the help rendered to me by various persons.

Firstly I thank God for showering his blessings on us. I am greatful to my institution City Engineering College for providing me an congenial atmosphere to carry out the project successfully.

I would like to express my heartful gratitude to **Dr. V S Ramamurthy, Principal**, CEC, Bangalore, for extending his support.

I would like to express my heartful gratitude to **Prof. Vivekavardhana Reddy**, HOD, Computer Science and Engineering whose guidance and support was truly invaluable.

I am very grateful to our guide, **Mrs. Ambika P R** Asst. Prof, Department of Computer Science, for her able guidance and valuable advice at every stage of our project which helped me in the successful completion of our project.

I would also have indebted to our Parent and Friends for their continued mora and material support throughout the course of project and helping me in finalize the presentation.

My heartly thanks to all those who have contributed bits, bytes and words to accomplish this project.

MEGHANA G(1CE17CS059)

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people.Although machine learning is a field within computer science, it differs from traditional computational approaches. In traditional computing, algorithms are sets of explicitly programmed instructions used by computers to calculate or problem solve. Machine learning algorithms instead allow for computers to train on data inputs and use statistical analysis in order to output values that fall within a specific range. Because of this, machine learning facilitates computers in building models from sample data in order to automate decision-making processes based on data inputs.

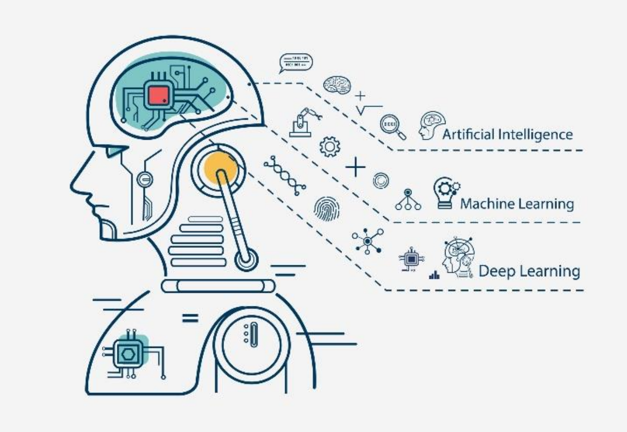


Fig 1.1: Introduction to ML

How does Machine Learning work:

Machine learning is a form of artificial intelligence (AI) that teaches computers to think in a similar way to how humans do learning and improving upon past experiences. It works by exploring data, identifying patterns, and involves minimal human intervention. Almost any task that can be completed with a data-defined pattern or set of rules can be automated with machine learning. This allows companies to transform processes that were previously only possible for humans to perform think responding to customer service calls, bookkeeping, and reviewing resumes.

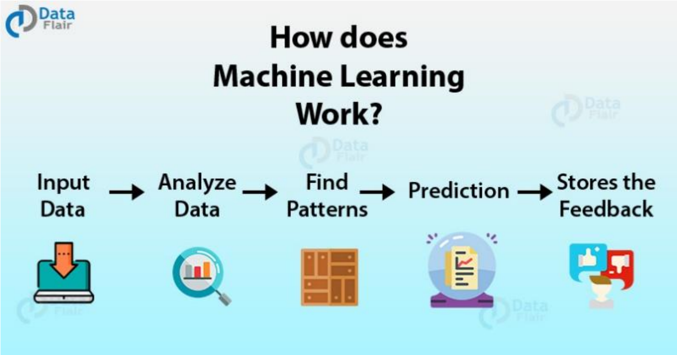


Fig 1.2: How does Machine Learning Work

Machine Learning Techniques:

Machine learning uses two types of techniques:

●**Supervised learning**: which trains a model on known input and output data so that it can predict future outputs. It allows you to collect data or produce a data output from a previous ML deployment. Supervised learning is exciting because it works in much the same way humans actually learn.

●**Unsupervised learning**: which finds hidden patterns or intrinsic structures in input data. helps you find all kinds of unknown patterns in data. In unsupervised learning, the algorithm tries to learn some inherent structure to the data with only unlabeled examples. Two common unsupervised learning tasks are clustering and dimensionality reduction.

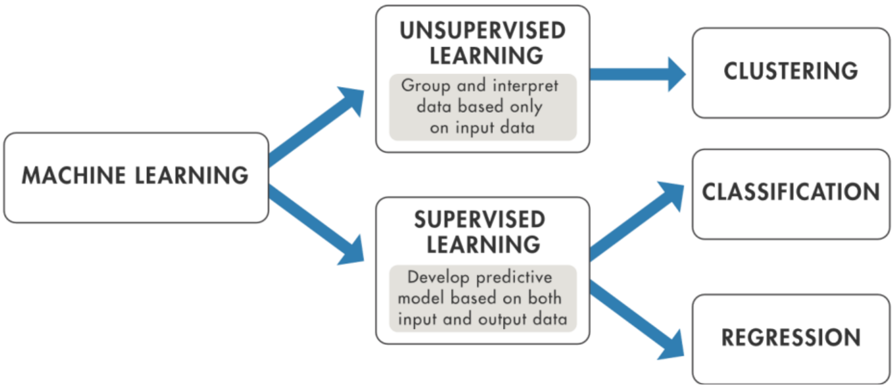

MACHINE LEARNING
UNSUPERVISED LEARNING
Group and interpret data based only on input data
SUPERVISED LEARNING
Develop predictive model based on both input and output data
CLUSTERING
CLASSIFICATION
REGRESSION

Fig 1.3: Techniques of Machine Learning

This report contains the “**REDWINE QUALITY PREDICTION**” based on the datset which contains the attributes such as fixed acidity, volatile acidity, citric acid, residual sugar,chlorides, free sulphuric acis, total sulphuric acid, density, pH, sulphates, alcohol, quantity.The Prediction is done through various machine learning algorithms such as

● Logistic Regression

● Support Vector Machine

● Decission Tree Classifier

● Random Forest Classifier

By using these algorithms we get a detailed view of the tested results and the trained results of the machine.We can visualize the results using Boxplot, Correlation matrix, jointplot and so on.

Objectives

The objectives of this project are as follows:

1.To experiment with different classification methods to see which yields the highest accuracy

2.To determine which features are the most indicative of a good quality wine

Chapter 2

TRAINING CONTENT

The Internship was successfully completed from the Tech Fortunes Technologies in the month of September. My guide for the internship training was Chetan Deepl who was a well versed teacher in the field of machine learning. The training took place in online mode and was carried out every day for 2 hours.

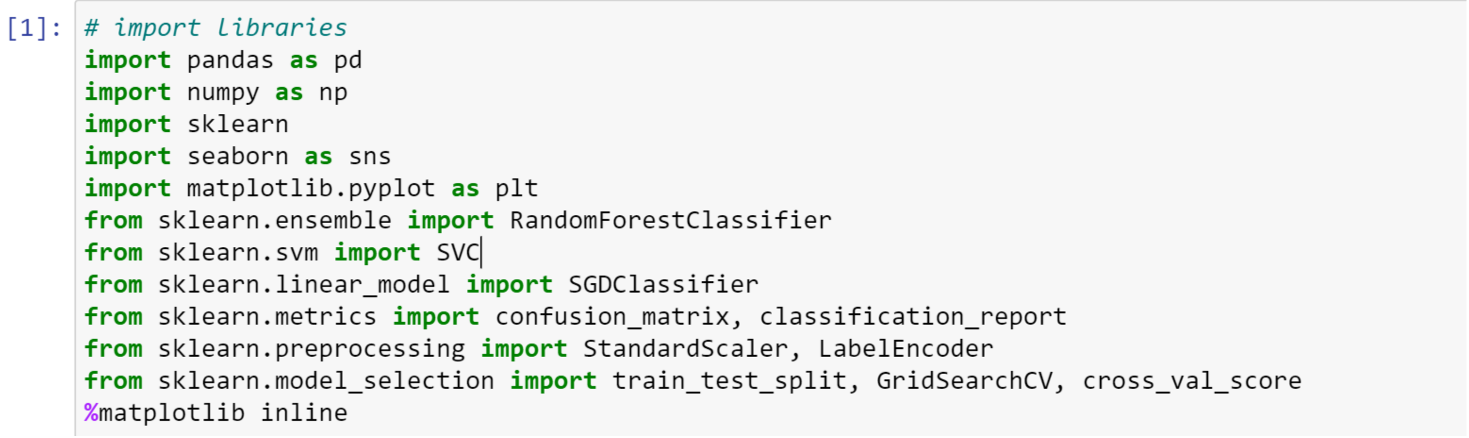
They firstly gave a brief introduction to the Python language as it was used for coding in machine learning. There were assignments given on the basics of python programs. Later they explained the Anaconda Jupyter Notebook, how it works, and its uses. There were many small projects that were taught like movie recommendation system, Iris flower Classification and many more. They taught us the Spider application in which we performed conversion of a image to white and black format, a live video in which we can change the colour of a persons clothes, while capturing a picture focusing on the face of the person, and so on.It is easy to debug the errors in Spyder applications as it identifies errors in each line. The different algorithms which we use for getting best accuracies like Linear Regression, K Means, Random Forest, SVM and many more were taught.

Overall it was a useful and a worthful internship program which provided me a lot of knowledge on Machine Learning its algorithms and how we have to estimate the training examples accuracy.

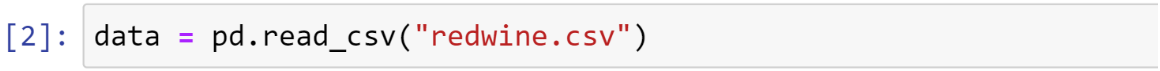
Chapter 3

IMPLEMENTATION

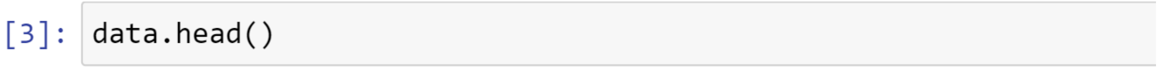
3.1 Source Code:



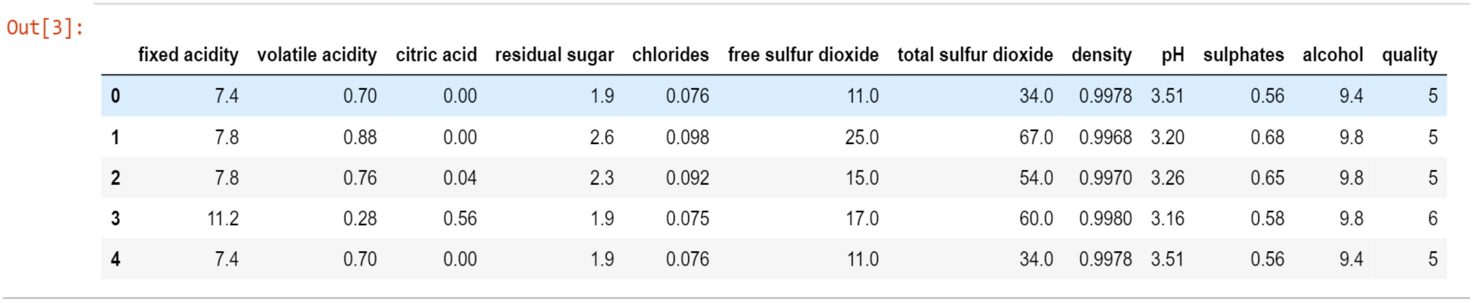
```
[1]: # import libraries
     import pandas as pd
     import numpy as np
     import sklearn
     import seaborn as sns
     import matplotlib.pyplot as plt
     from sklearn.ensemble import RandomForestClassifier
     from sklearn.svm import SVC
     from sklearn.linear_model import SGDClassifier
     from sklearn.metrics import confusion_matrix, classification_report
     from sklearn.preprocessing import StandardScaler, LabelEncoder
     from sklearn.model_selection import train_test_split, GridSearchCV, cross_val_score
     %matplotlib inline
```



```
[2]: data = pd.read_csv("redwine.csv")
```

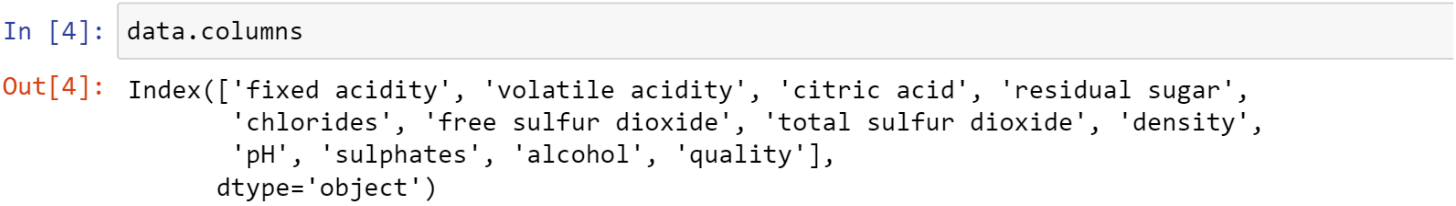


```
[3]: data.head()
```

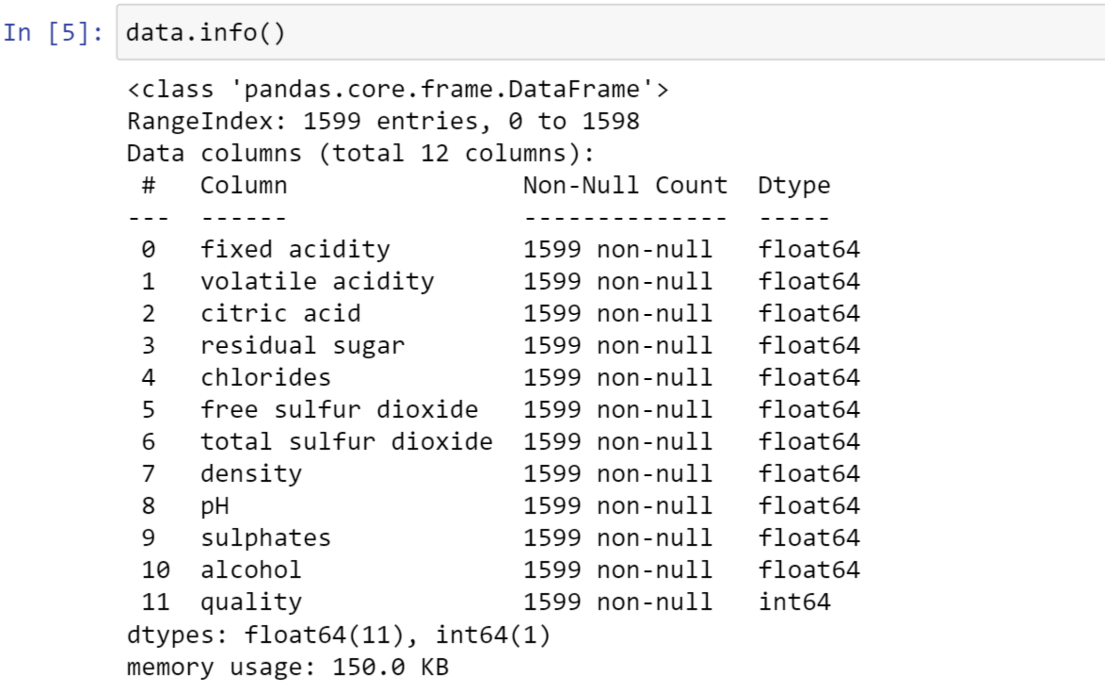


Out[3]:

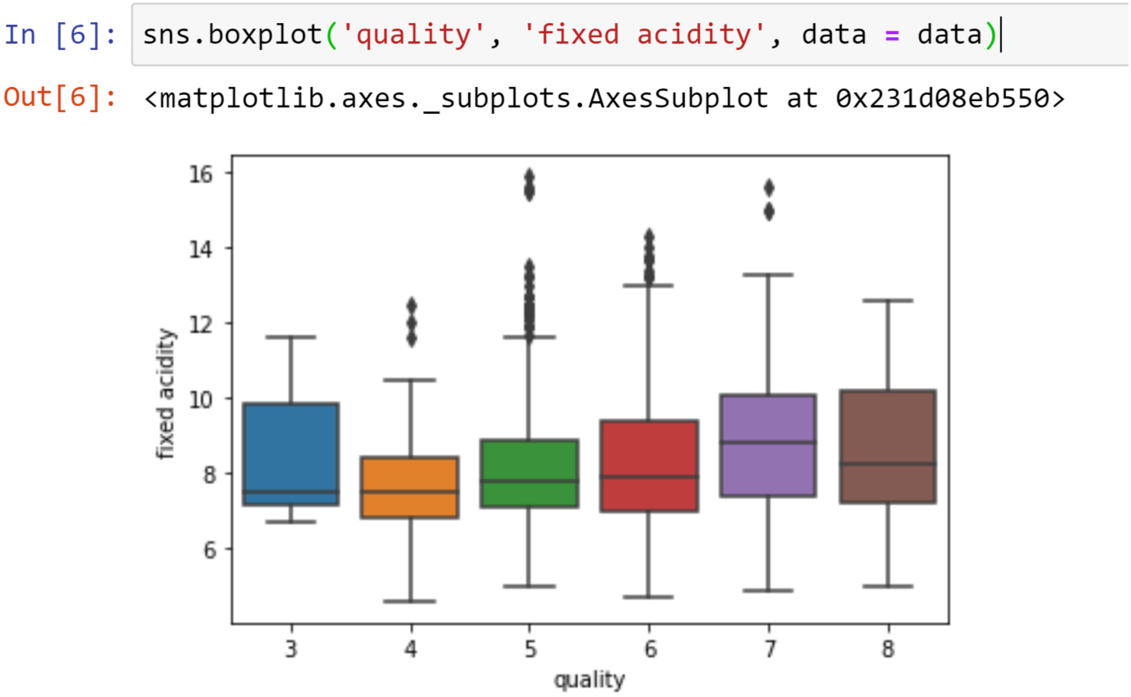
	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	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



```
In [4]: data.columns

Out[4]: Index(['fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar',
               'chlorides', 'free sulfur dioxide', 'total sulfur dioxide', 'density',
               'pH', 'sulphates', 'alcohol', 'quality'],
              dtype='object')
```

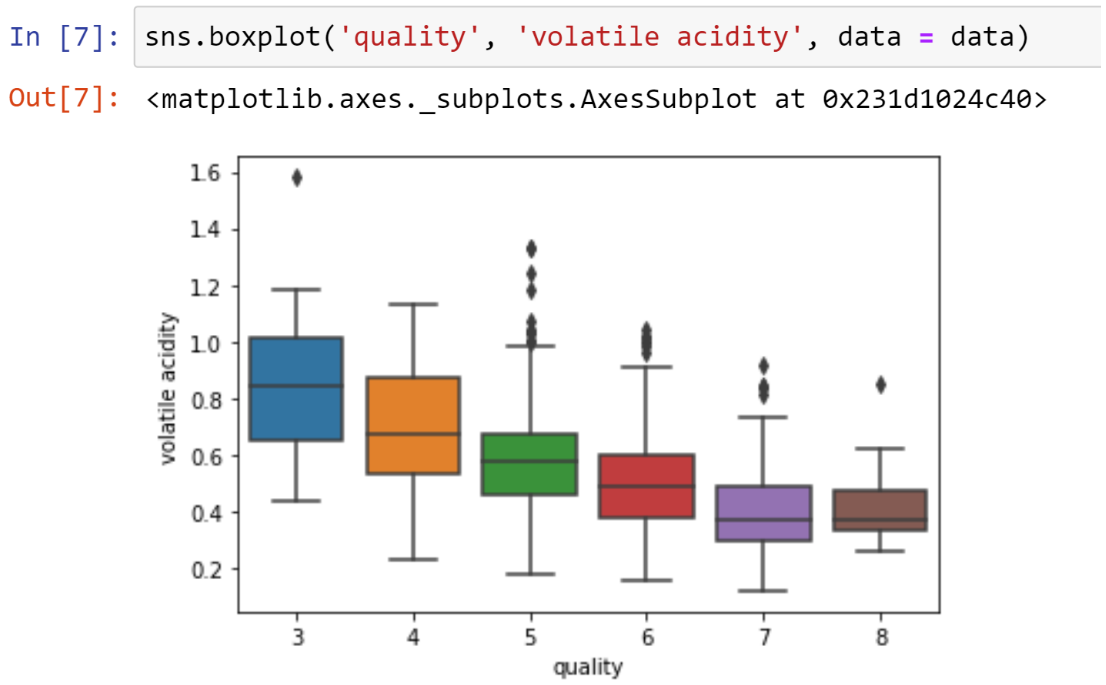


```
In [5]: data.info()

<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
```



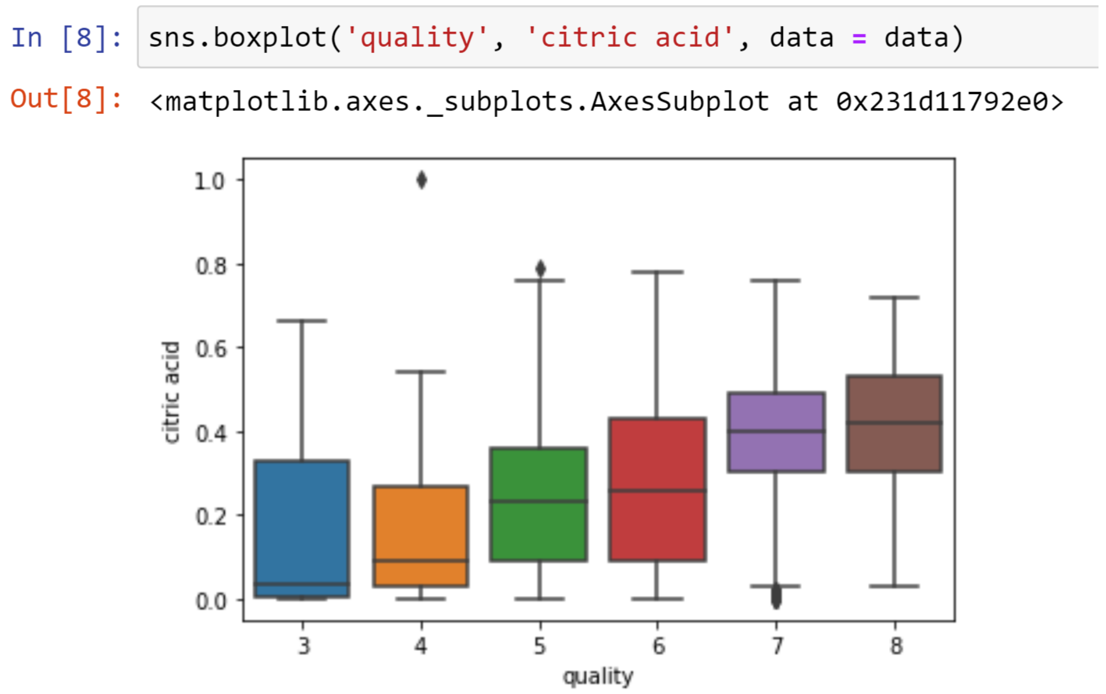
```
In [6]: sns.boxplot('quality', 'fixed acidity', data = data)
Out[6]: <matplotlib.axes._subplots.AxesSubplot at 0x231d08eb550>
```

Visualization: Here we see that fixed acidity does not give any specification to classify the quality.



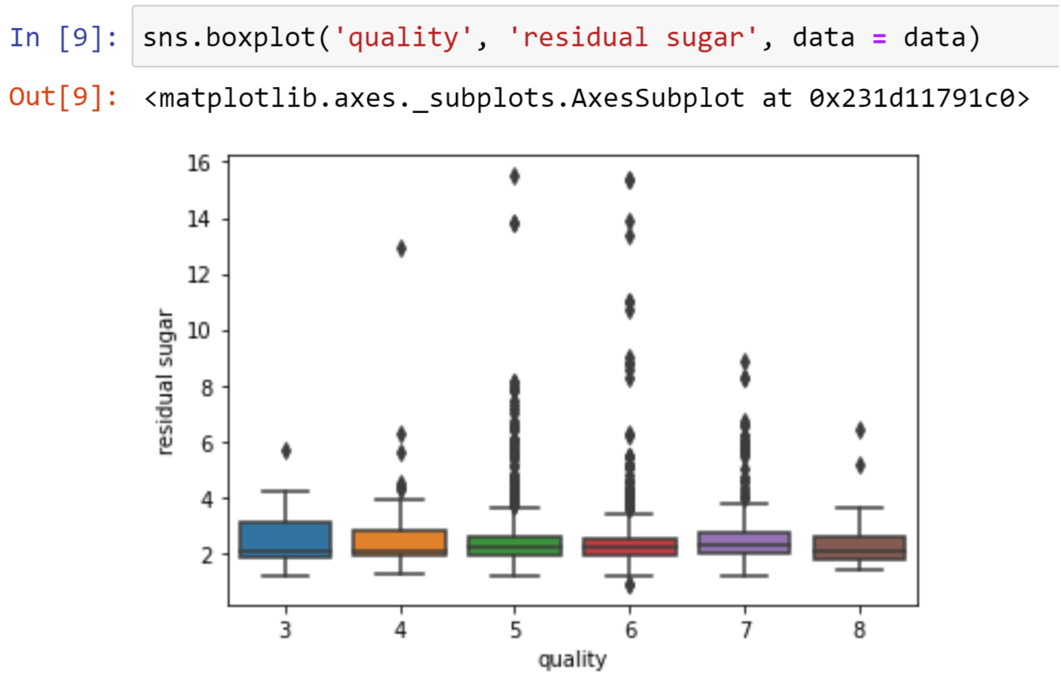
```
In [7]: sns.boxplot('quality', 'volatile acidity', data = data)
Out[7]: <matplotlib.axes._subplots.AxesSubplot at 0x231d1024c40>
```

Visualization: Here we see that its quite a downing trend in the volatile acidity as we go higher the quality.



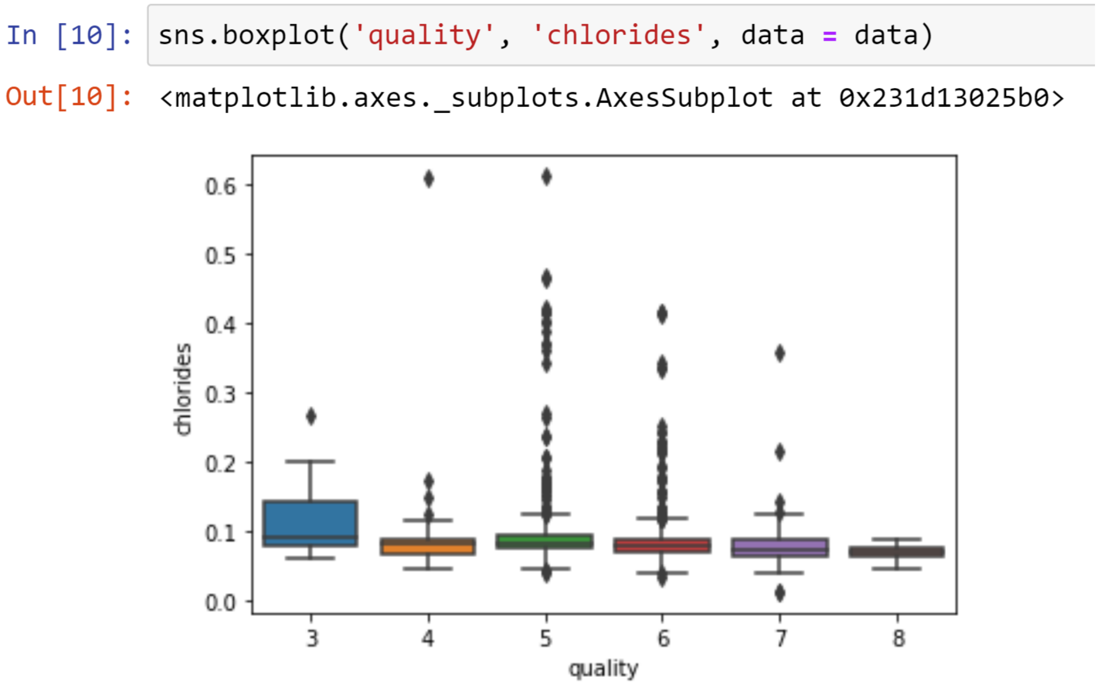
```
In [8]: sns.boxplot('quality', 'citric acid', data = data)
Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x231d11792e0>
```

Visualization: Composition of citric acid goes higher as we go higher in the quality of the wine.



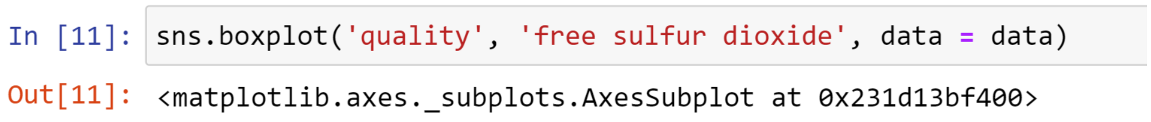
```
In [9]: sns.boxplot('quality', 'residual sugar', data = data)

Out[9]: <matplotlib.axes._subplots.AxesSubplot at 0x231d11791c0>
```

Visualization: Composition of residual sugar is uniformly distributed over the different quality level.

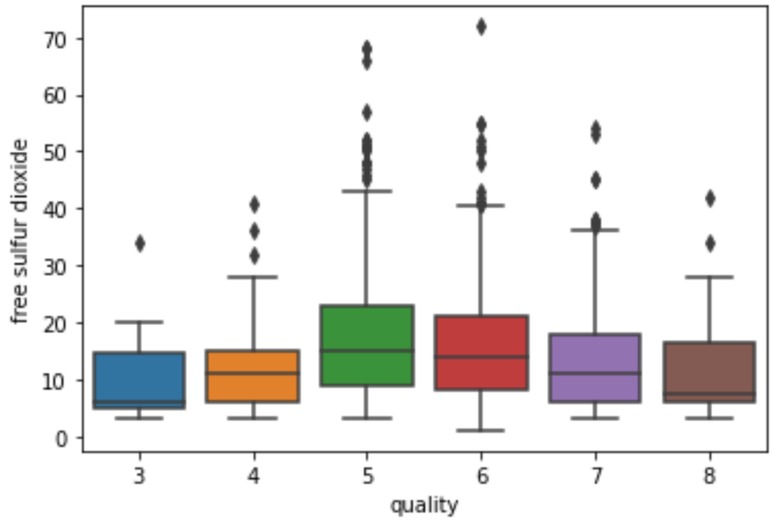


```
In [10]: sns.boxplot('quality', 'chlorides', data = data)

Out[10]: <matplotlib.axes._subplots.AxesSubplot at 0x231d13025b0>
```

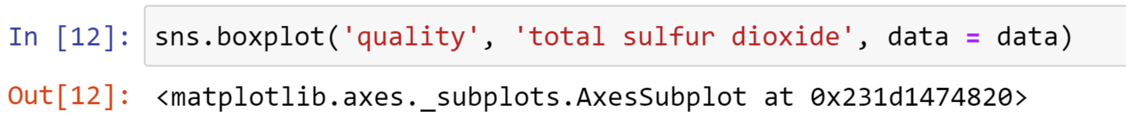
Visualization: Composition of chlorides also goes down as we go higher in the quality of the wine.



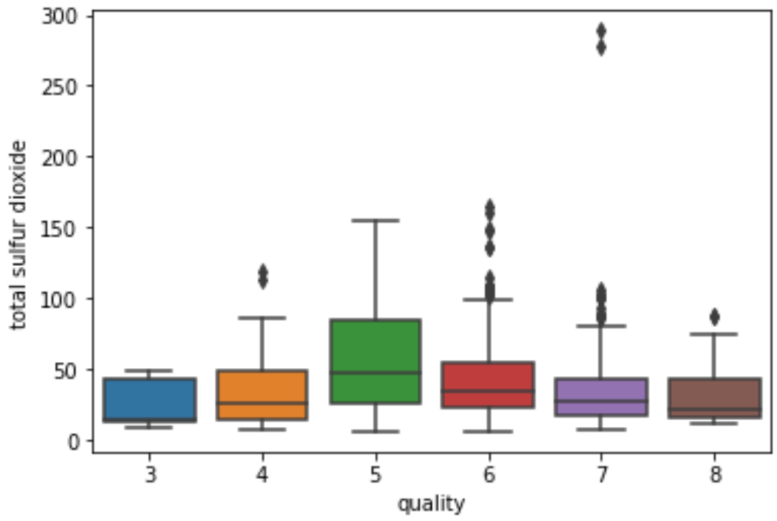
```
In [11]: sns.boxplot('quality', 'free sulfur dioxide', data = data)
Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x231d13bf400>
```



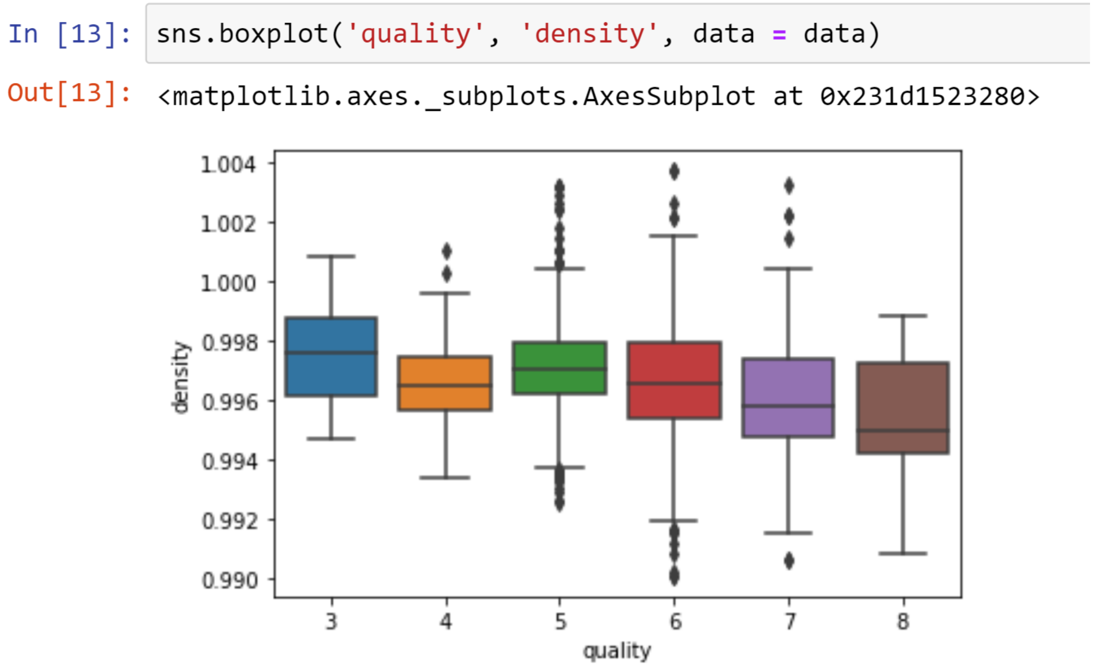
Visualization: Composition of free sulphur dioxide somewhat increases a bit in the quality levels of 5 and 6 and then again decreases.



```
In [12]: sns.boxplot('quality', 'total sulfur dioxide', data = data)
Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0x231d1474820>
```

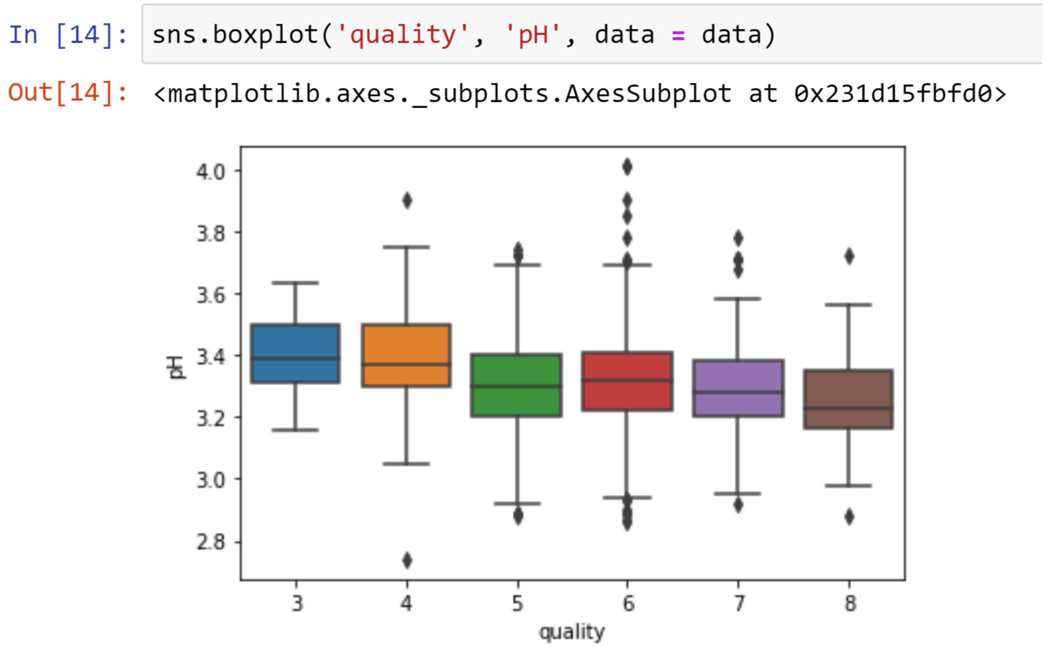


Visualization: Composition of total sulphur dioxide lowers towards the high quality levels with an sudden increase in the quality level of 5.



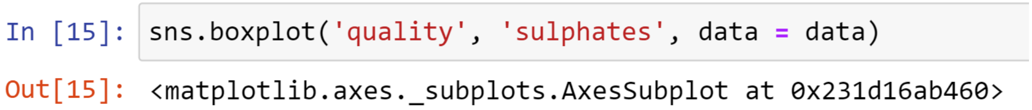
```
In [13]: sns.boxplot('quality', 'density', data = data)
Out[13]: <matplotlib.axes._subplots.AxesSubplot at 0x231d1523280>
```

Visualization: Density of the red wine seems to be larger at the low quality levels and tends to decrease towards the high quality levels.

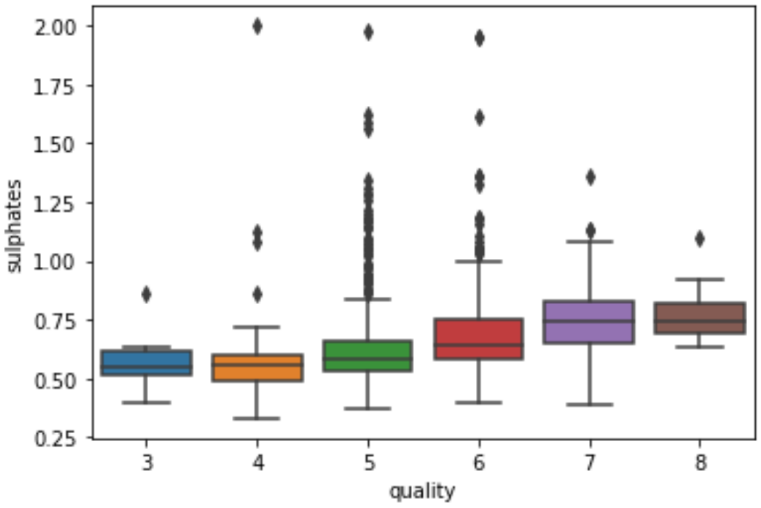


```
In [14]: sns.boxplot('quality', 'pH', data = data)
Out[14]: <matplotlib.axes._subplots.AxesSubplot at 0x231d15fbfd0>
```

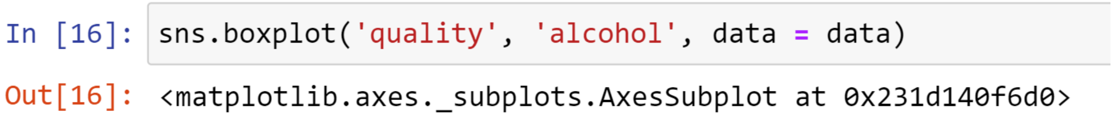
Visualization: The pH levels of the red wine seems to be uniformly distributed with the quality levels with the little increase in the pH at the quality level 3 and 4.



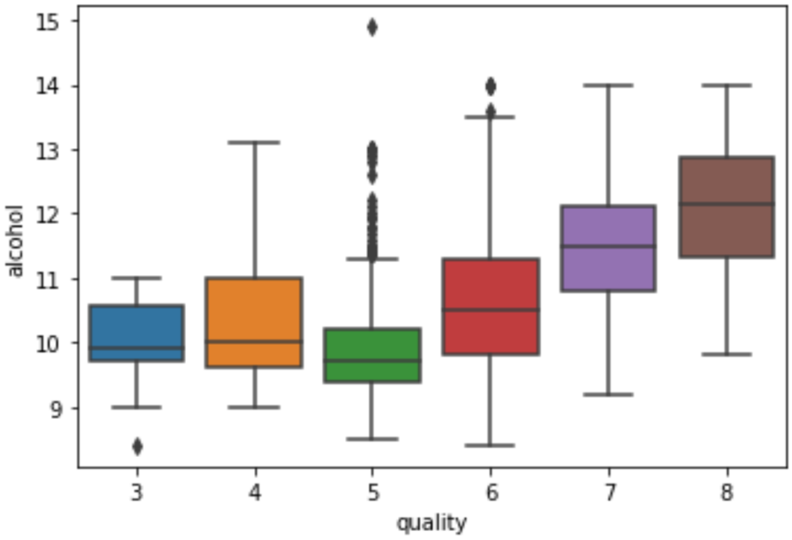
```
In [15]: sns.boxplot('quality', 'sulphates', data = data)

Out[15]: <matplotlib.axes._subplots.AxesSubplot at 0x231d16ab460>
```



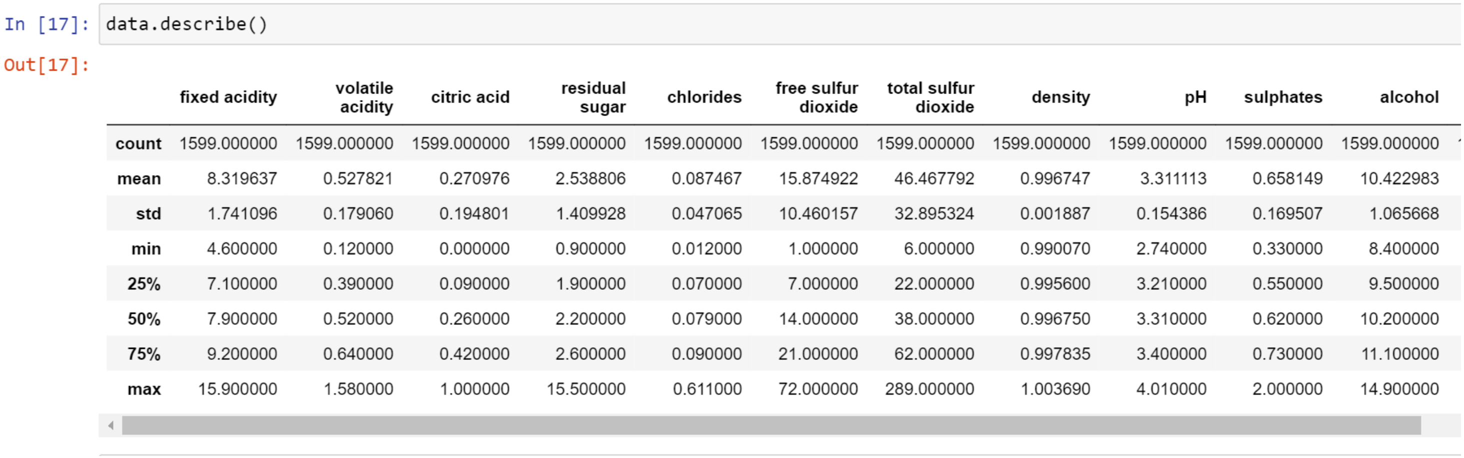
Visualization: Sulphates level goes higher with the quality of wine.



```
In [16]: sns.boxplot('quality', 'alcohol', data = data)

Out[16]: <matplotlib.axes._subplots.AxesSubplot at 0x231d140f6d0>
```

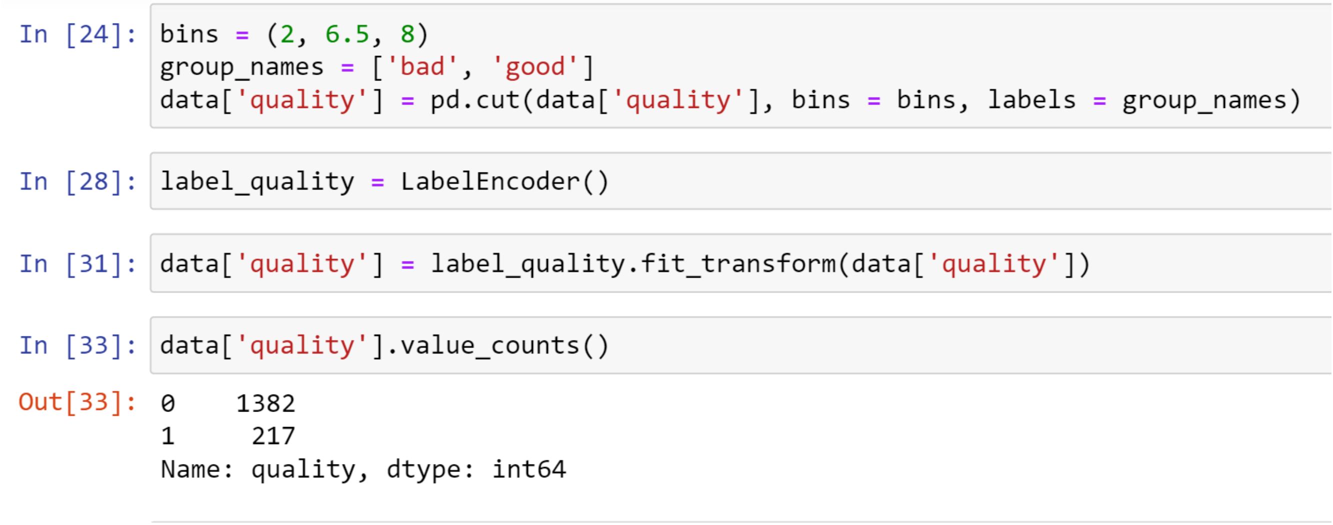


Visualization: Alcohol level also goes higher as the quality of wine increases.

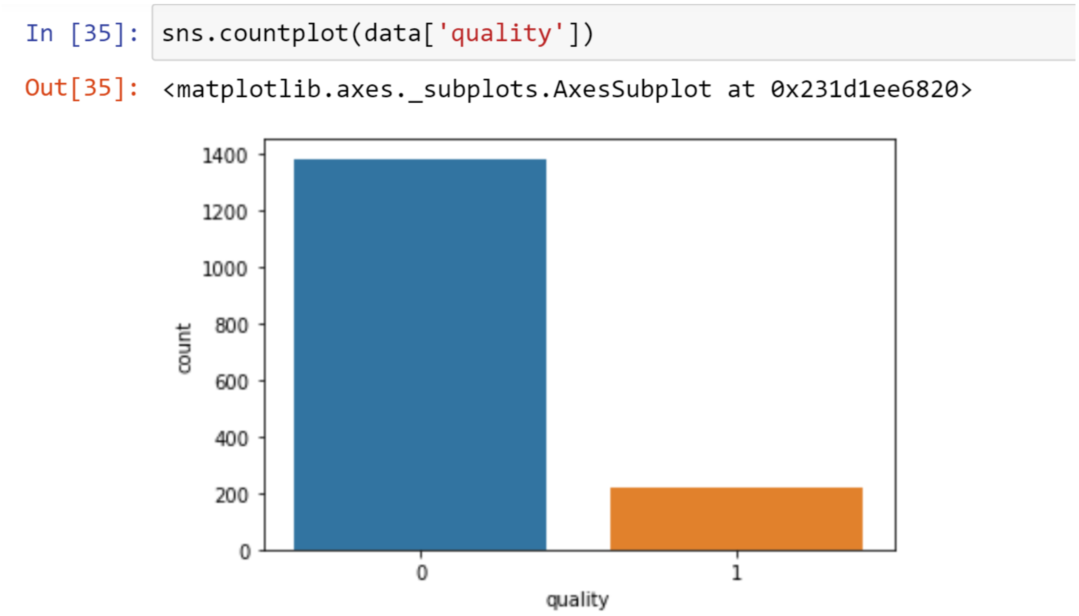


```
In [17]: data.describe()
Out[17]:
```

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol
count	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000
mean	8.319637	0.527821	0.270976	2.538806	0.087467	15.874922	46.467792	0.996747	3.311113	0.658149	10.422983
std	1.741096	0.179060	0.194801	1.409928	0.047065	10.460157	32.895324	0.001887	0.154386	0.169507	1.065668
min	4.600000	0.120000	0.000000	0.900000	0.012000	1.000000	6.000000	0.990070	2.740000	0.330000	8.400000
25%	7.100000	0.390000	0.090000	1.900000	0.070000	7.000000	22.000000	0.995600	3.210000	0.550000	9.500000
50%	7.900000	0.520000	0.260000	2.200000	0.079000	14.000000	38.000000	0.996750	3.310000	0.620000	10.200000
75%	9.200000	0.640000	0.420000	2.600000	0.090000	21.000000	62.000000	0.997835	3.400000	0.730000	11.100000
max	15.900000	1.580000	1.000000	15.500000	0.611000	72.000000	289.000000	1.003690	4.010000	2.000000	14.900000

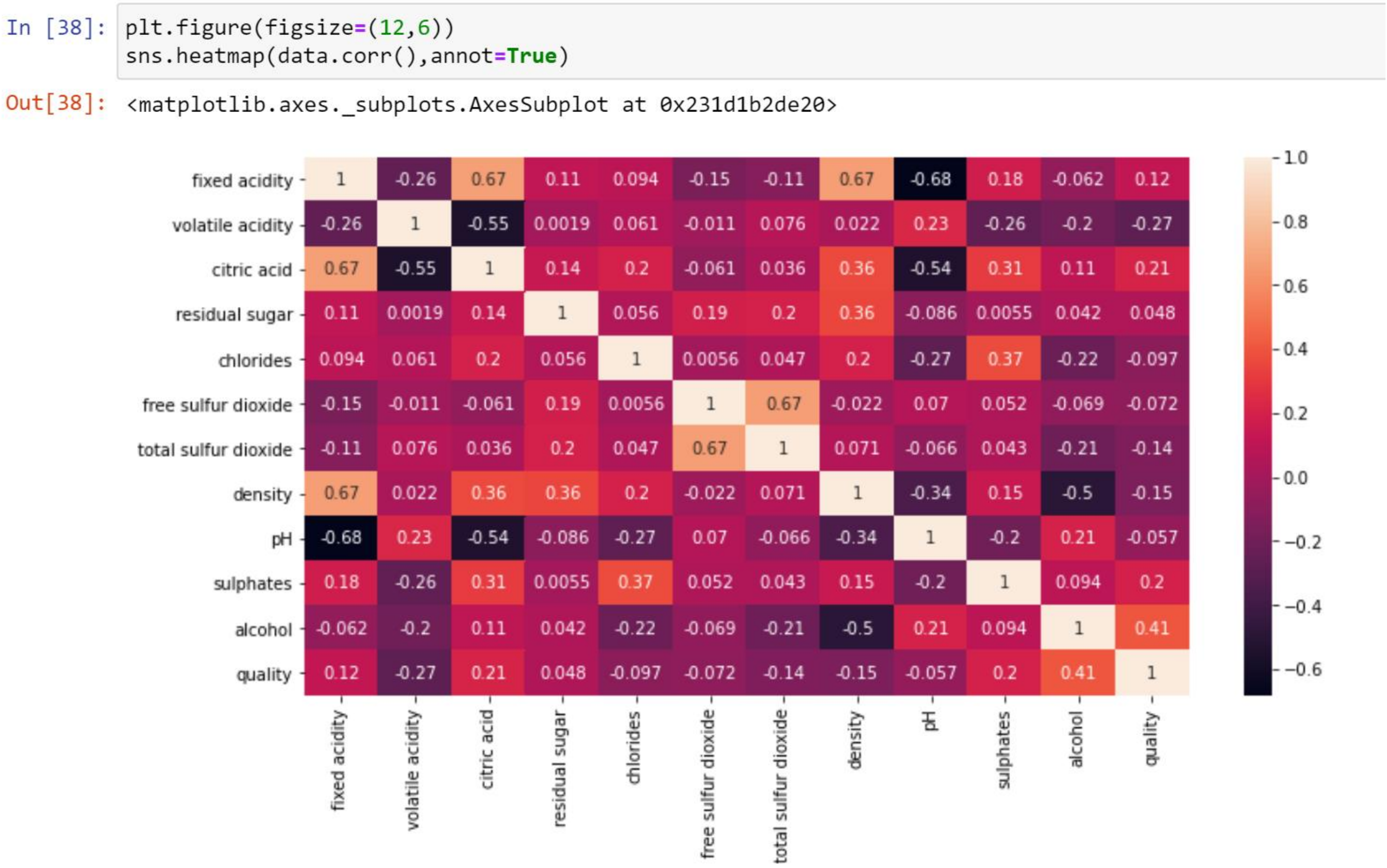


```
In [24]: bins = (2, 6.5, 8)
         group_names = ['bad', 'good']
         data['quality'] = pd.cut(data['quality'], bins = bins, labels = group_names)

In [28]: label_quality = LabelEncoder()

In [31]: data['quality'] = label_quality.fit_transform(data['quality'])

In [33]: data['quality'].value_counts()

Out[33]: 0    1382
         1     217
         Name: quality, dtype: int64
```



```
In [35]: sns.countplot(data['quality'])

Out[35]: <matplotlib.axes._subplots.AxesSubplot at 0x231d1ee6820>
```

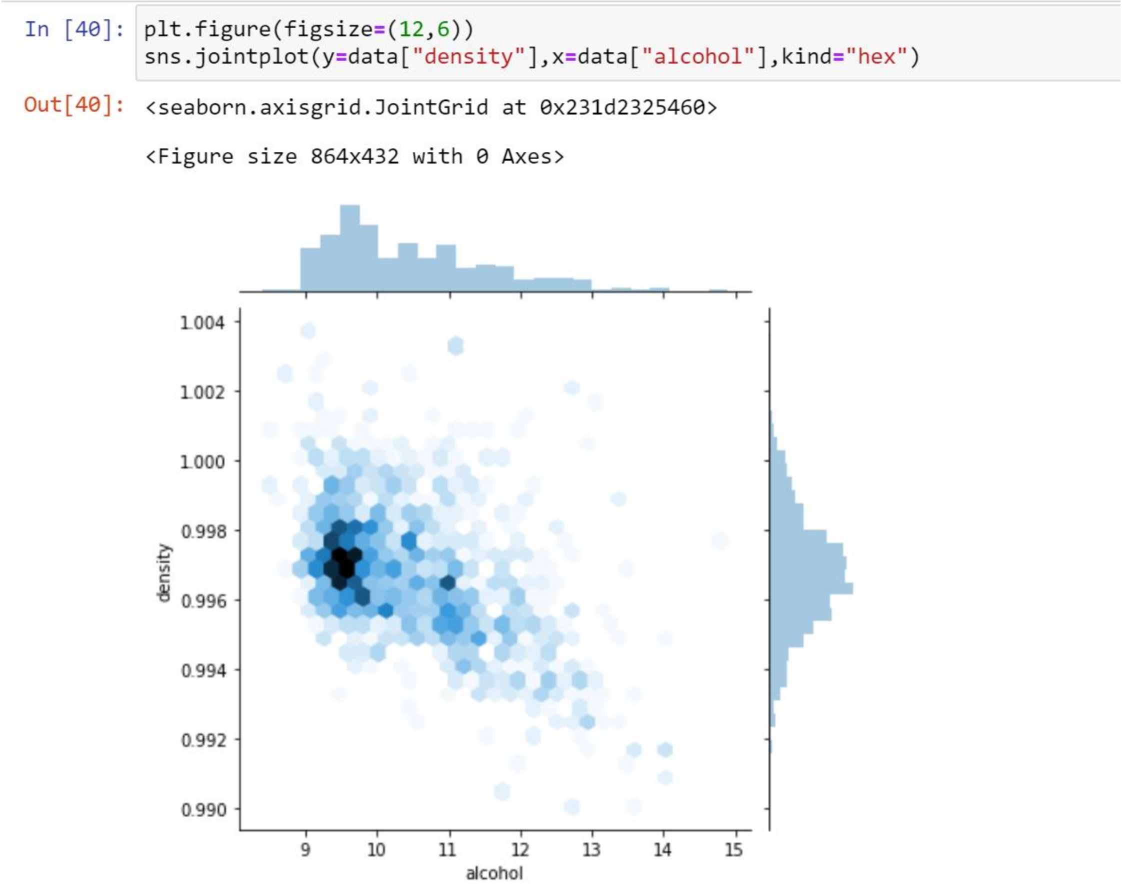
Visualization: From the above countplot we can see that there are large no. of wines with good quality=0 means whose quality levels are less than 7.



```
In [38]: plt.figure(figsize=(12,6))
         sns.heatmap(data.corr(),annot=True)
```

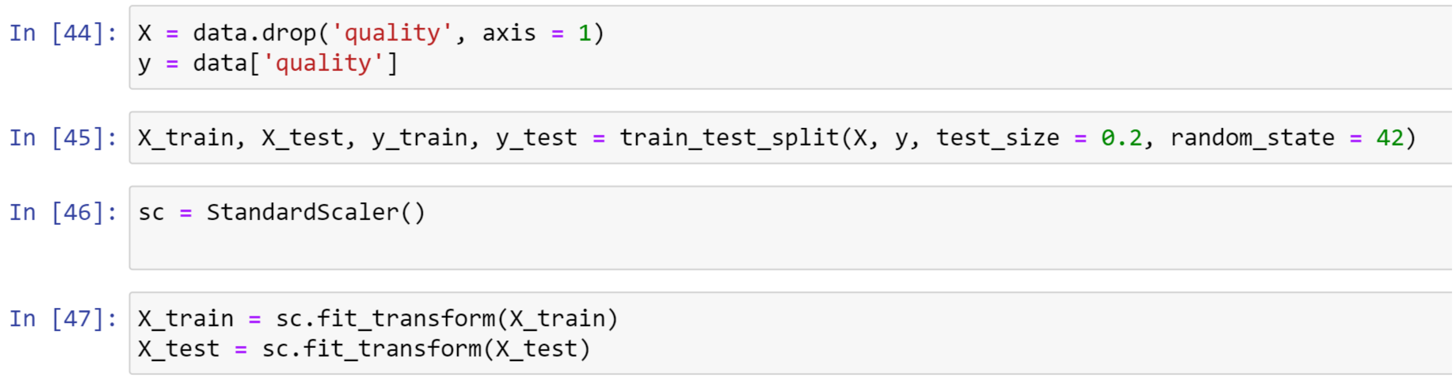
```
Out[38]: <matplotlib.axes._subplots.AxesSubplot at 0x231d1b2de20>
```

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
fixed acidity	1	-0.26	0.67	0.11	0.094	-0.15	-0.11	0.67	-0.68	0.18	-0.062	0.12
volatile acidity	-0.26	1	-0.55	0.0019	0.061	-0.011	0.076	0.022	0.23	-0.26	-0.2	-0.27
citric acid	0.67	-0.55	1	0.14	0.2	-0.061	0.036	0.36	-0.54	0.31	0.11	0.21
residual sugar	0.11	0.0019	0.14	1	0.056	0.19	0.2	0.36	-0.086	0.0055	0.042	0.048
chlorides	0.094	0.061	0.2	0.056	1	0.0056	0.047	0.2	-0.27	0.37	-0.22	-0.097
free sulfur dioxide	-0.15	-0.011	-0.061	0.19	0.0056	1	0.67	-0.022	0.07	0.052	-0.069	-0.072
total sulfur dioxide	-0.11	0.076	0.036	0.2	0.047	0.67	1	0.071	-0.066	0.043	-0.21	-0.14
density	0.67	0.022	0.36	0.36	0.2	-0.022	0.071	1	-0.34	0.15	-0.5	-0.15
pH	-0.68	0.23	-0.54	-0.086	-0.27	0.07	-0.066	-0.34	1	-0.2	0.21	-0.057
sulphates	0.18	-0.26	0.31	0.0055	0.37	0.052	0.043	0.15	-0.2	1	0.094	0.2
alcohol	-0.062	-0.2	0.11	0.042	-0.22	-0.069	-0.21	-0.5	0.21	0.094	1	0.41
quality	0.12	-0.27	0.21	0.048	-0.097	-0.072	-0.14	-0.15	-0.057	0.2	0.41	1

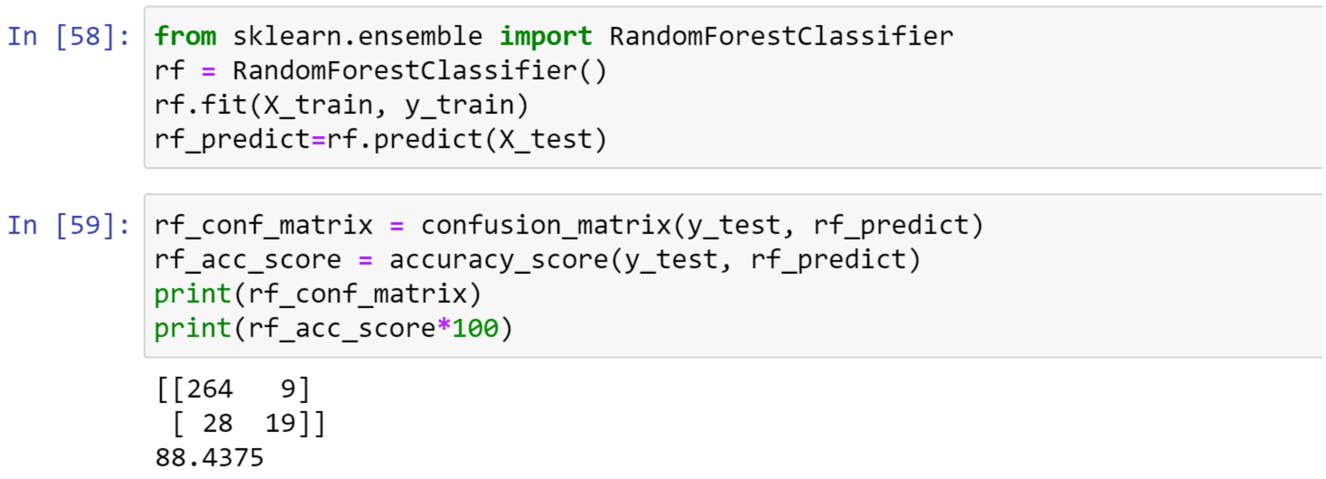


```
In [40]: plt.figure(figsize=(12,6))
         sns.jointplot(y=data["density"],x=data["alcohol"],kind="hex")
```

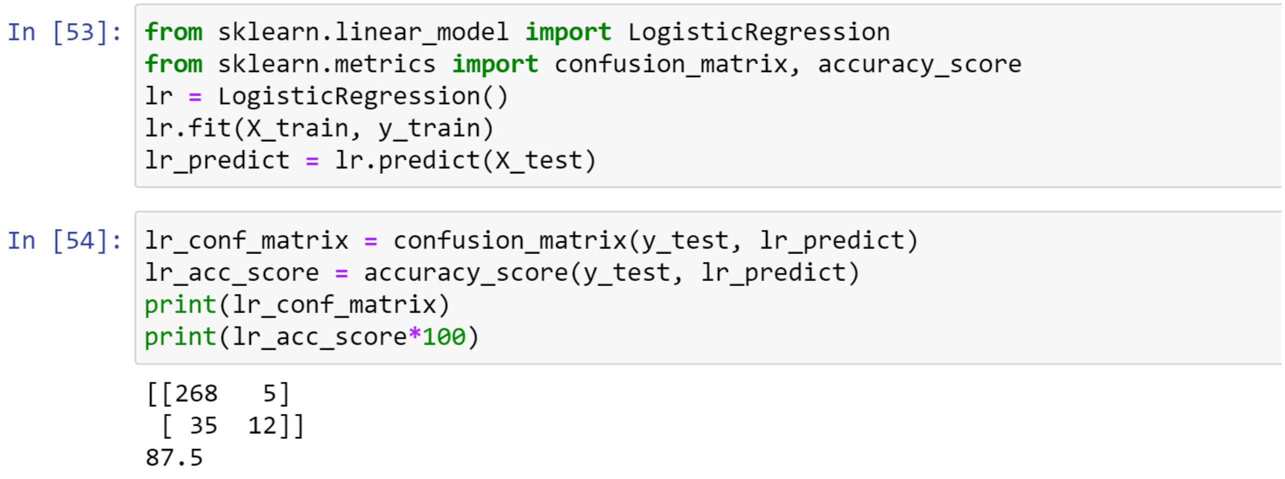
```
Out[40]: <seaborn.axisgrid.JointGrid at 0x231d2325460>

         <Figure size 864x432 with 0 Axes>
```



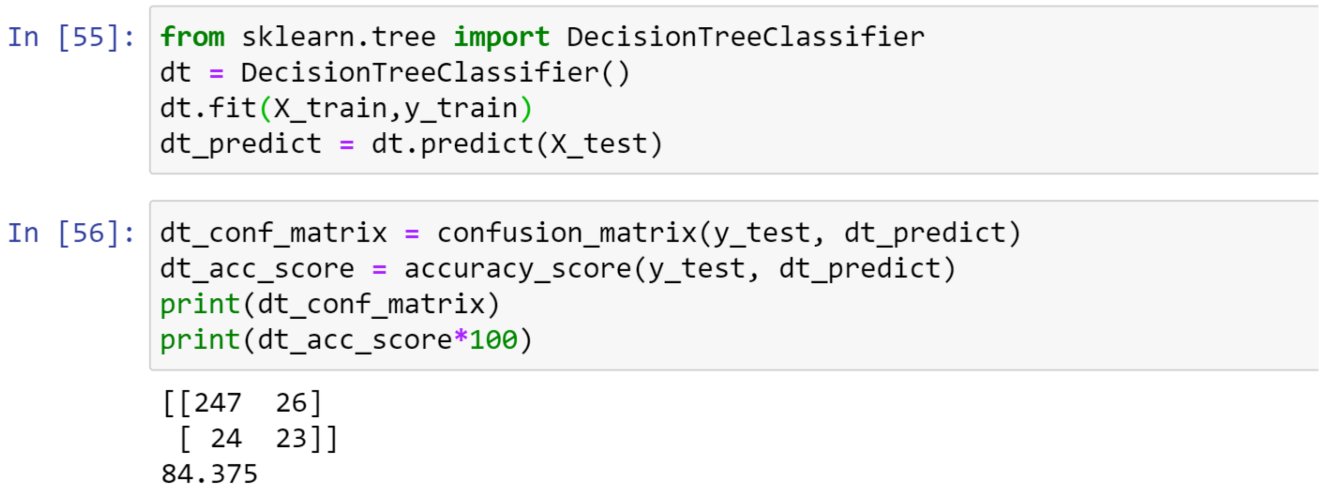
```
In [44]: X = data.drop('quality', axis = 1)
         y = data['quality']

In [45]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 42)

In [46]: sc = StandardScaler()

In [47]: X_train = sc.fit_transform(X_train)
         X_test = sc.fit_transform(X_test)
```



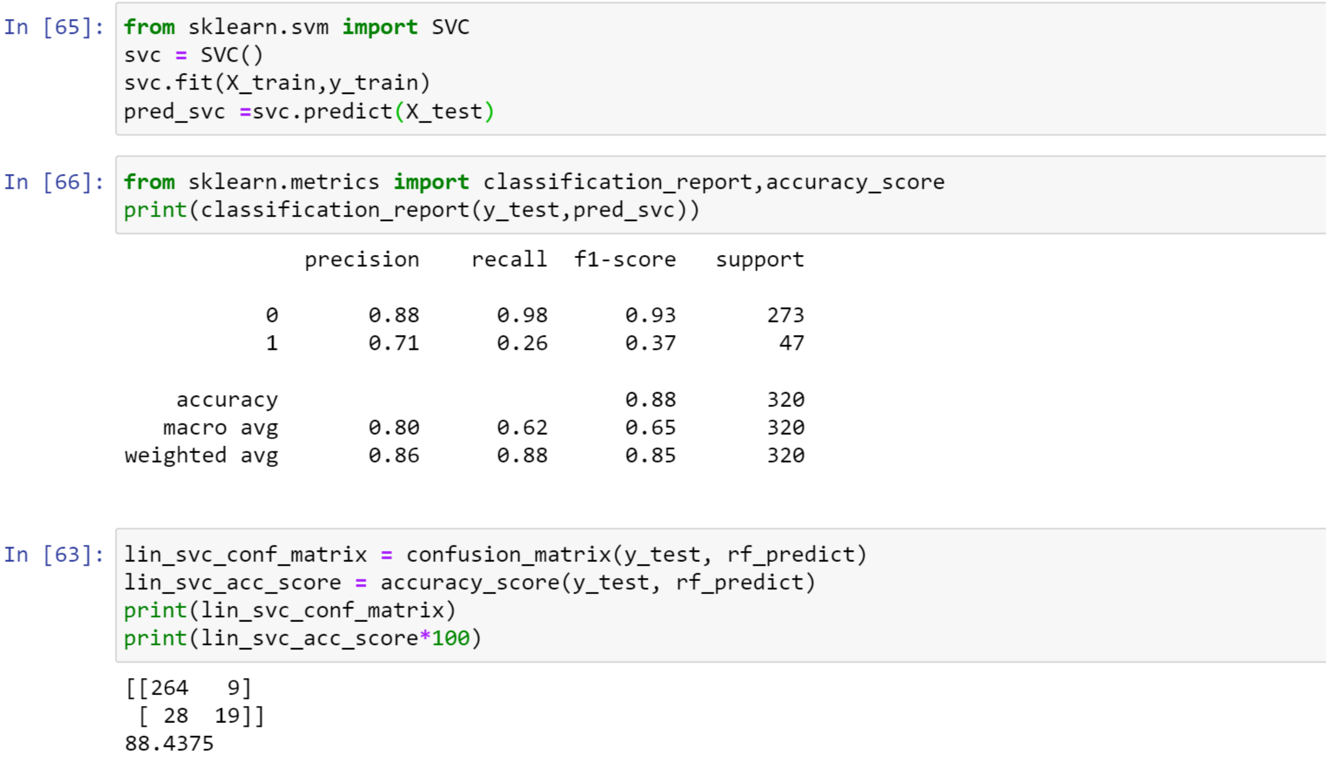
```
In [58]: from sklearn.ensemble import RandomForestClassifier
         rf = RandomForestClassifier()
         rf.fit(X_train, y_train)
         rf_predict=rf.predict(X_test)

In [59]: rf_conf_matrix = confusion_matrix(y_test, rf_predict)
         rf_acc_score = accuracy_score(y_test, rf_predict)
         print(rf_conf_matrix)
         print(rf_acc_score*100)

         [[264   9]
          [ 28  19]]
         88.4375
```



```
In [53]: from sklearn.linear_model import LogisticRegression
         from sklearn.metrics import confusion_matrix, accuracy_score
         lr = LogisticRegression()
         lr.fit(X_train, y_train)
         lr_predict = lr.predict(X_test)

In [54]: lr_conf_matrix = confusion_matrix(y_test, lr_predict)
         lr_acc_score = accuracy_score(y_test, lr_predict)
         print(lr_conf_matrix)
         print(lr_acc_score*100)

         [[268   5]
          [ 35  12]]
         87.5
```



```
In [55]: from sklearn.tree import DecisionTreeClassifier
         dt = DecisionTreeClassifier()
         dt.fit(X_train,y_train)
         dt_predict = dt.predict(X_test)

In [56]: dt_conf_matrix = confusion_matrix(y_test, dt_predict)
         dt_acc_score = accuracy_score(y_test, dt_predict)
         print(dt_conf_matrix)
         print(dt_acc_score*100)

         [[247  26]
          [ 24  23]]
         84.375
```



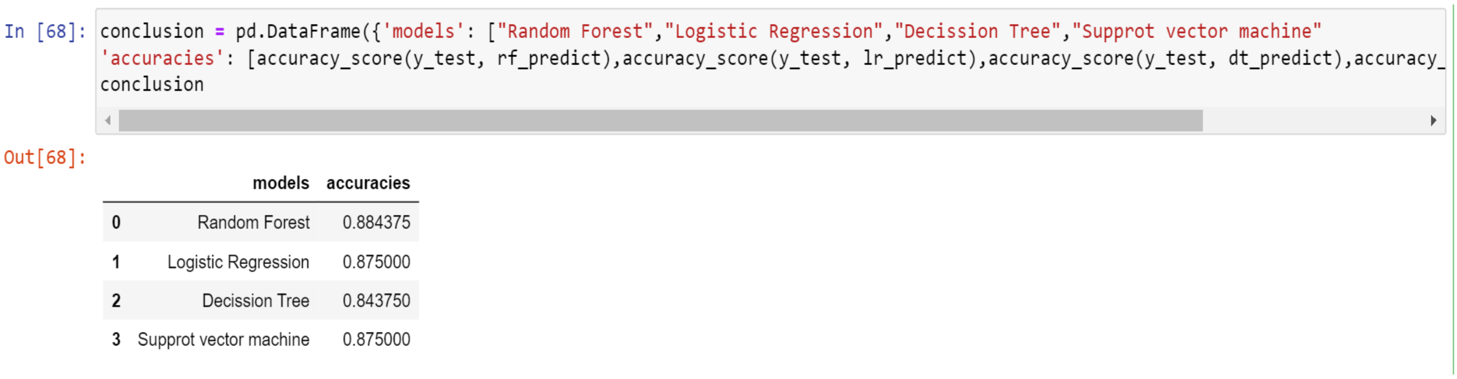
```
In [65]: from sklearn.svm import SVC
         svc = SVC()
         svc.fit(X_train,y_train)
         pred_svc =svc.predict(X_test)
```

```
In [66]: from sklearn.metrics import classification_report,accuracy_score
         print(classification_report(y_test,pred_svc))
```

```
              precision    recall  f1-score   support

           0       0.88      0.98      0.93       273
           1       0.71      0.26      0.37        47

    accuracy                           0.88       320
   macro avg       0.80      0.62      0.65       320
weighted avg       0.86      0.88      0.85       320
```

```
In [63]: lin_svc_conf_matrix = confusion_matrix(y_test, rf_predict)
         lin_svc_acc_score = accuracy_score(y_test, rf_predict)
         print(lin_svc_conf_matrix)
         print(lin_svc_acc_score*100)
```

```
[[264   9]
 [ 28  19]]
88.4375
```



```
In [68]: conclusion = pd.DataFrame({'models': ["Random Forest","Logistic Regression","Decission Tree","Supprot vector machine"
         'accuracies': [accuracy_score(y_test, rf_predict),accuracy_score(y_test, lr_predict),accuracy_score(y_test, dt_predict),accuracy_
         conclusion
```

Out[68]:

	models	accuracies
0	Random Forest	0.884375
1	Logistic Regression	0.875000
2	Decission Tree	0.843750
3	Supprot vector machine	0.875000

Chapter 4

LEARNING OUTCOMES

I have understood the various equipment's working principle, company details, specifications and distributors details. The whole internship period has motivated me to design a system, component, or process to meet desired needs with realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability. It made me understand the function on multidisciplinary teams and inspired me to create a novel system to solve engineering problems. It has made me Understand professional and ethical responsibility and to Communicate effectively. I have obtained the broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.

From All the Classification Algorithms we can see that the **Random Forest Classifier** algorithm which yields heighest Accuracy of **88.5 %.** Along with Random Forest Classifier, SVM, Decision tree and Logistic Regression also gives a good Accuracy of 87%.

DECLARATION

We the students of 7th semester BE, Computer Science and Engineering hereby declare that project entitled “RedWine Quality Prediction” has been carried out by us at City Engineering College, Bengaluru and submitted in partial fulfilment of the cource requirement for the award of the degree of **Bachelor of Engineerirng in computer Science and Engineering of Visvesvaraya Technological University,Belgaum,** during the academic year 2019-2020.

I also declare that, to the best of the knowledge and belief ,the work reported here does not form the part of dissertation on the basis of which a degree or award was conferred on a earlier occasion on this by any other student.

Date :

Place: Bangalore

MEGHANA G

(1CE17CS059)