**Analysis of Wine sample through application of ML classification algorithms**

Written by

* Shashank Shankar(18UCS135)
* Hardik jain(18UCC056)
* Kumar Raghav Sharma(18UCC156)
* Chaduvula Aporv(18UCC137)

**Introduction**

In our project, we worked on the dataset relating to red Vinho Werde wine. Our goal is to model, analyze and predict wine quality through classifications.

**Libraries used**

1. Pandas
2. Numpy
3. Matplotlib
4. Seaborn
   * Countplot
   * Heat Map
   * Box Plot
   * Factor Plot
5. Pandas\_profiling
6. Sklearn

* Train\_test\_split
* Standard Scaler
* KNeighborsClassifier
* Cross\_val\_score
* Metrics
* DecisionTreeClassifier

**Dataset description**

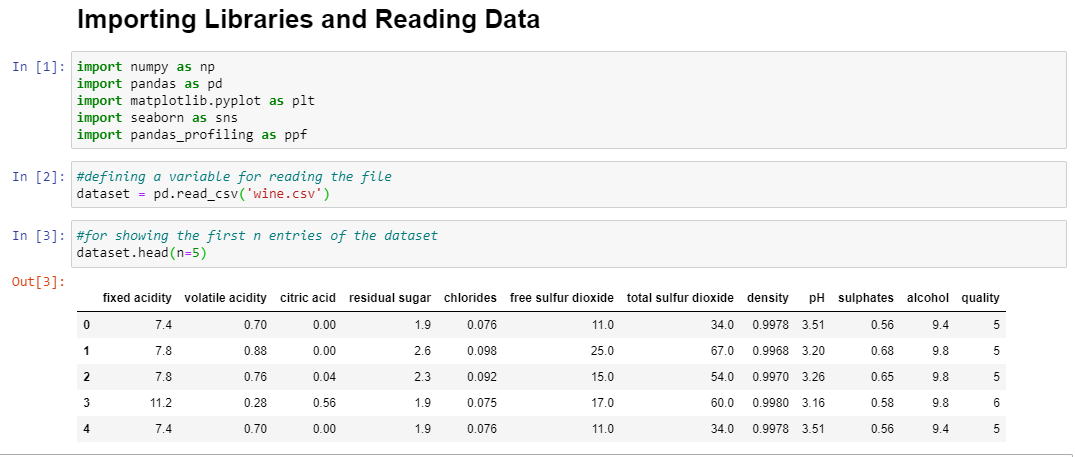
Wine sample dataset has been taken from the following database.

<https://archive.ics.uci.edu/ml/datasets/Wine+Quality>

The dataset comprises of 1599 tuples, and each tuple is characterized by 12 attributes.

**Data Overview**

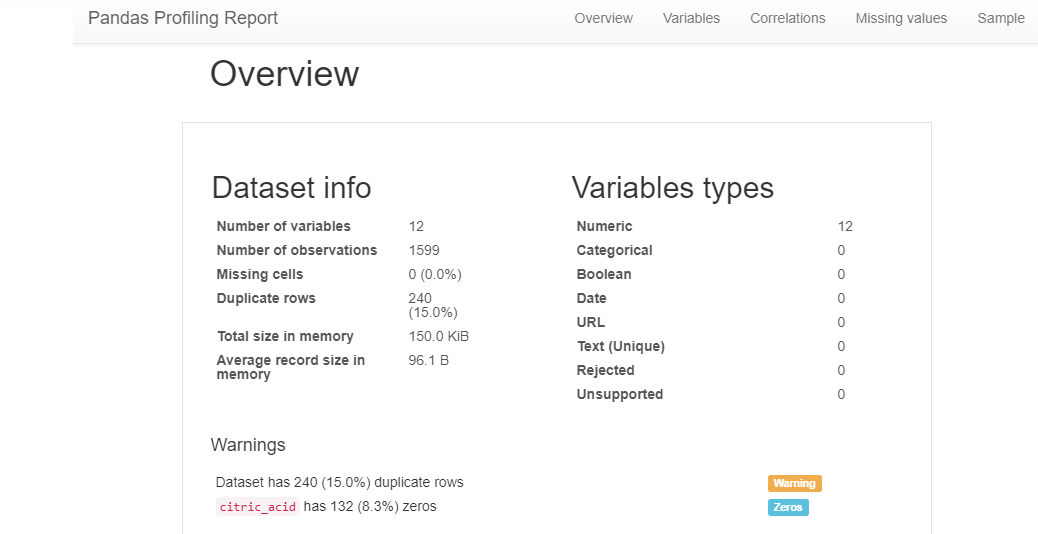
Summary of the attributes has been provided below.

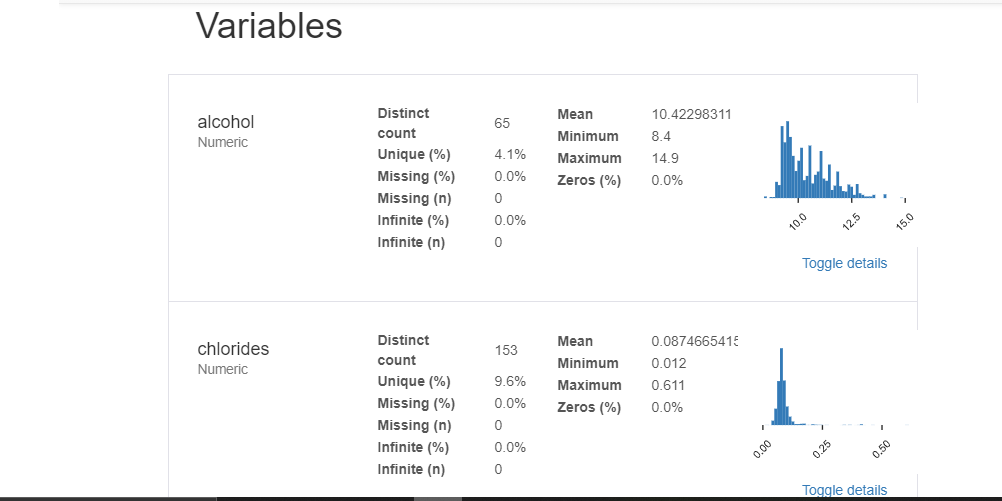


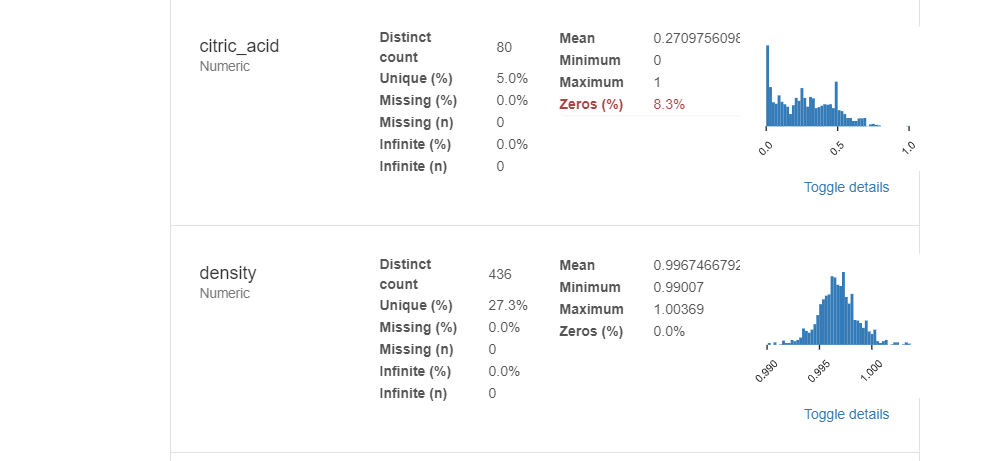
* Libraries are imported and read\_csv() is used to read the file.
* In-built function head() is used to display the first n entries of the database.

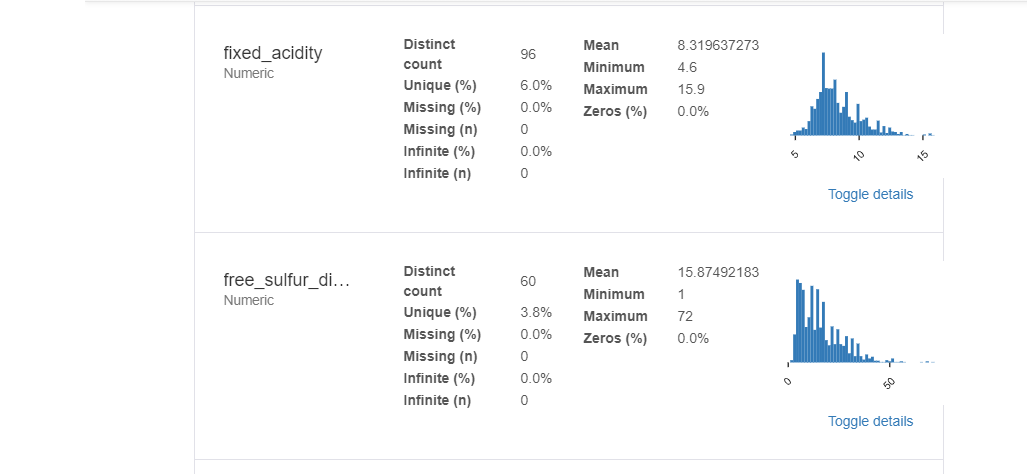
The pandas-profiling Python package is a great tool to create HTML profiling reports. Such reports provide good overview of the data.

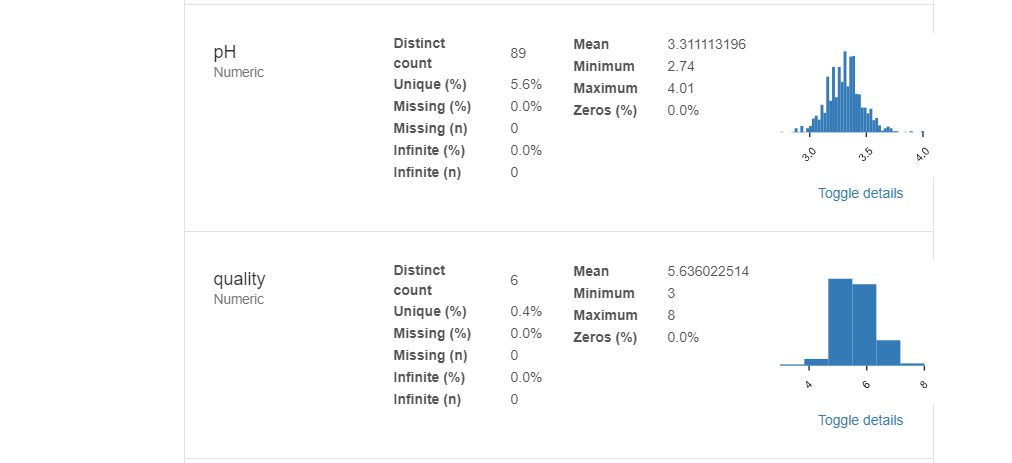


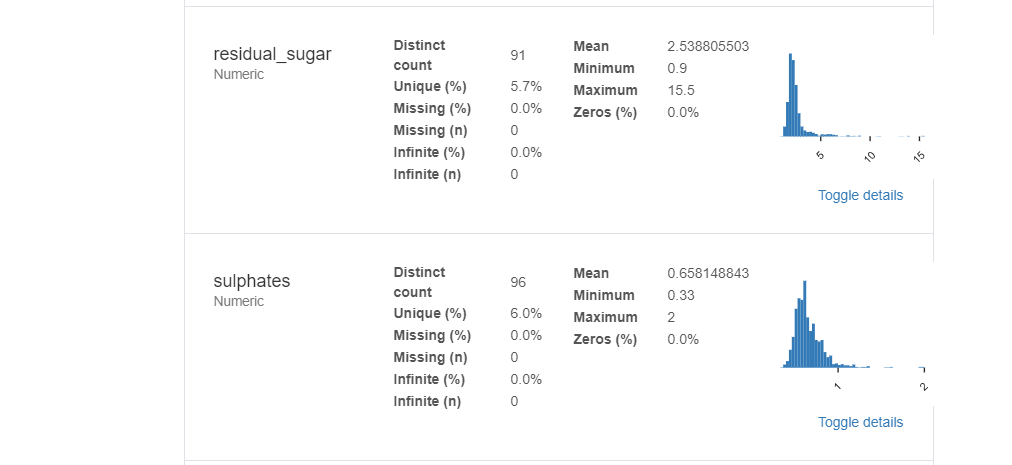


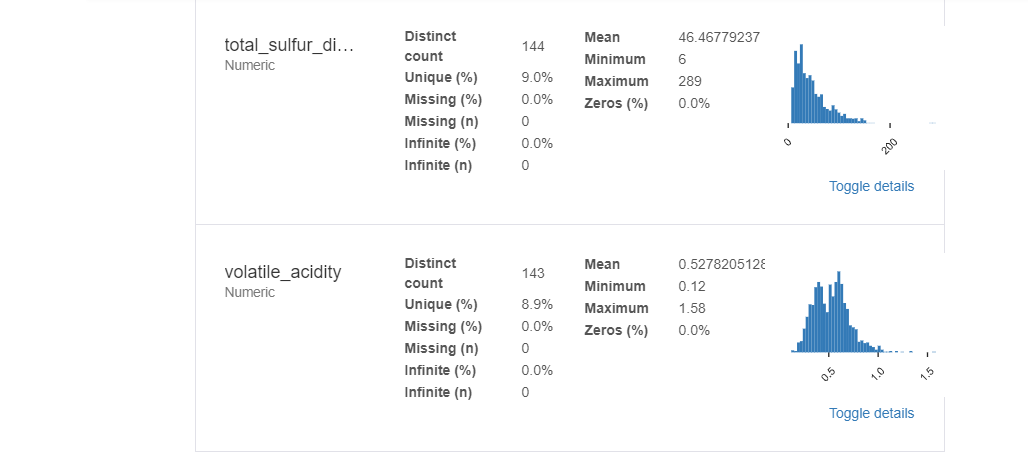




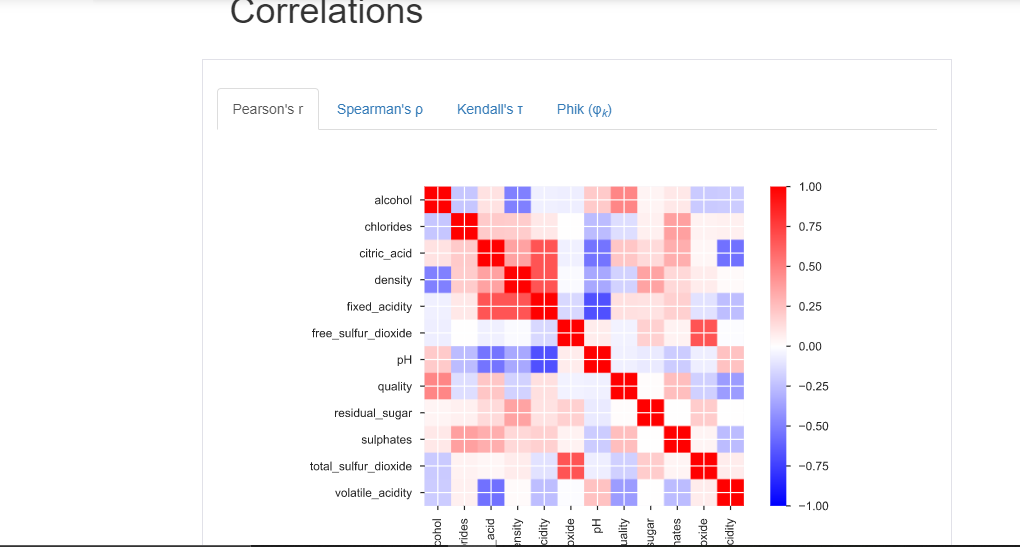


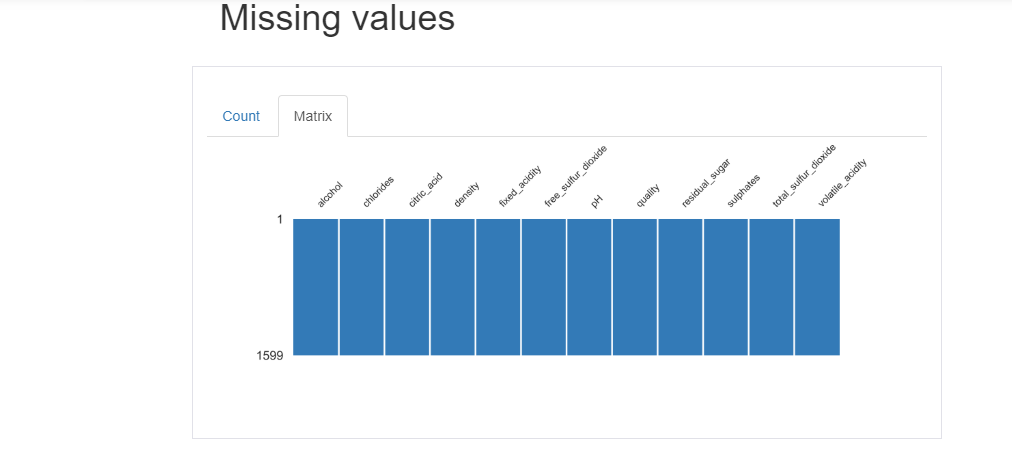


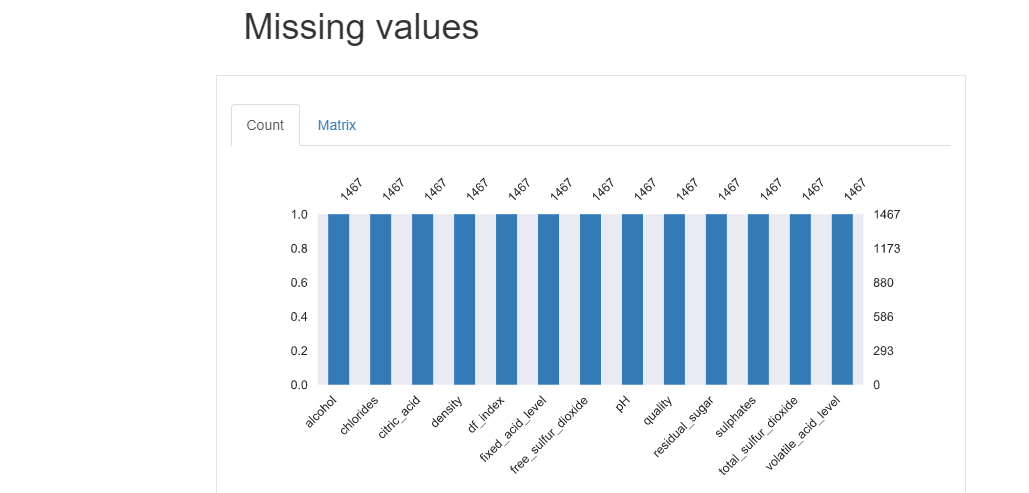




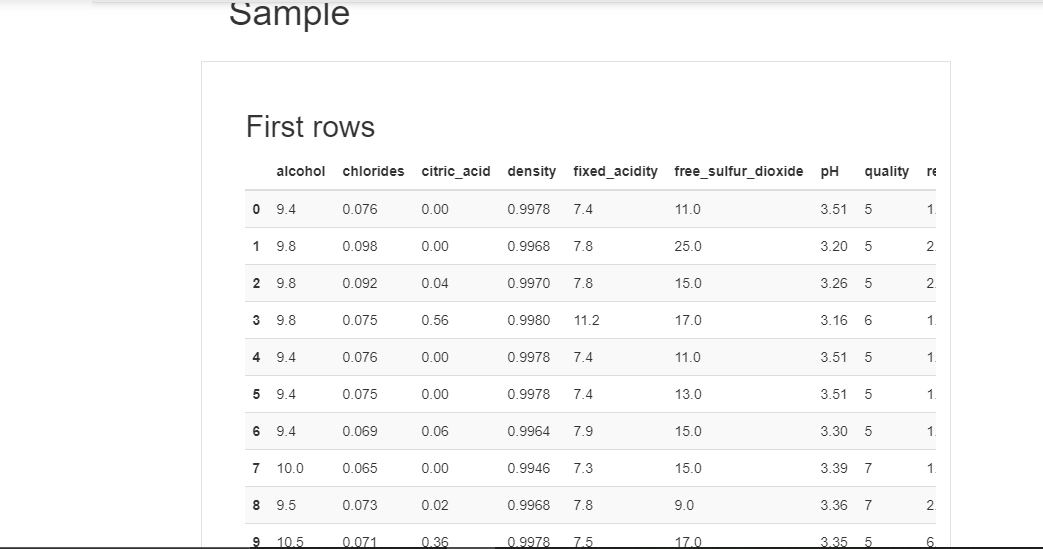
The correlation between attributes is being mapped through heat maps.

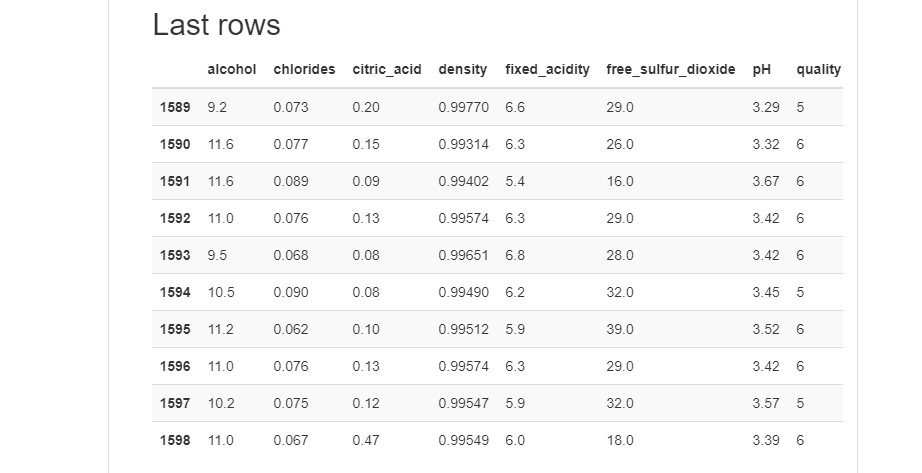






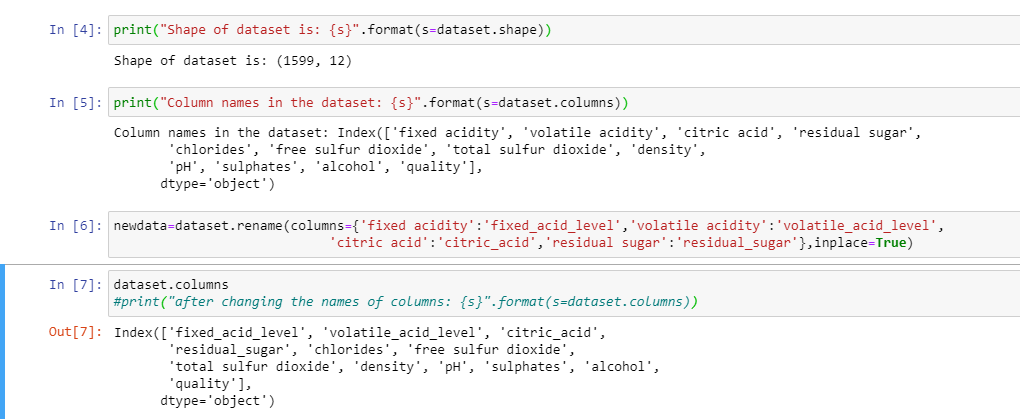
Samples of the dataset are provided below.



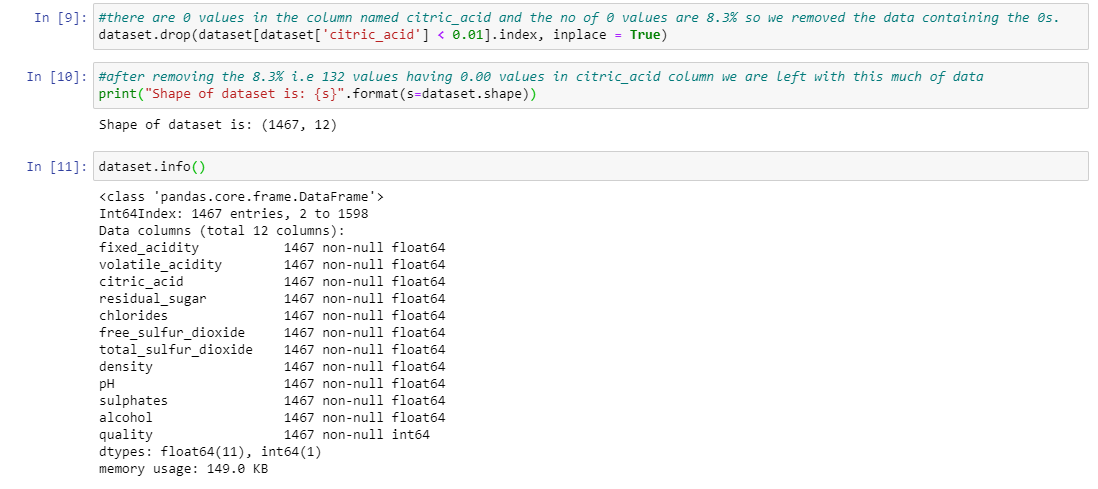


**Data studying and cleaning**

Here, first we renamed attributes and remove spaces for ease of use.

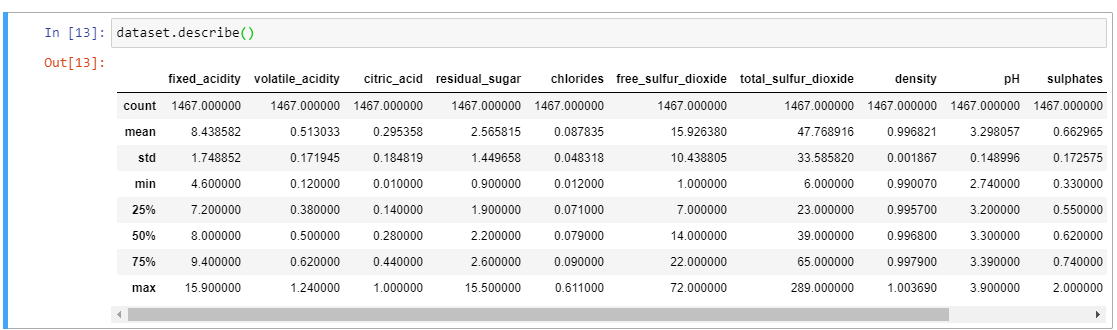


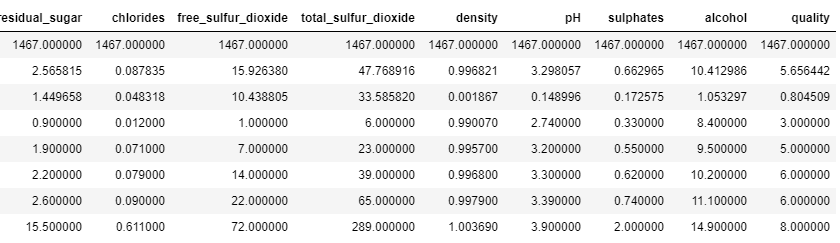
Tuples with citric\_acid value as 0 are removed to obtain cleaner data.





The describe() method is used to provide a summary of statistical measures as listed below.

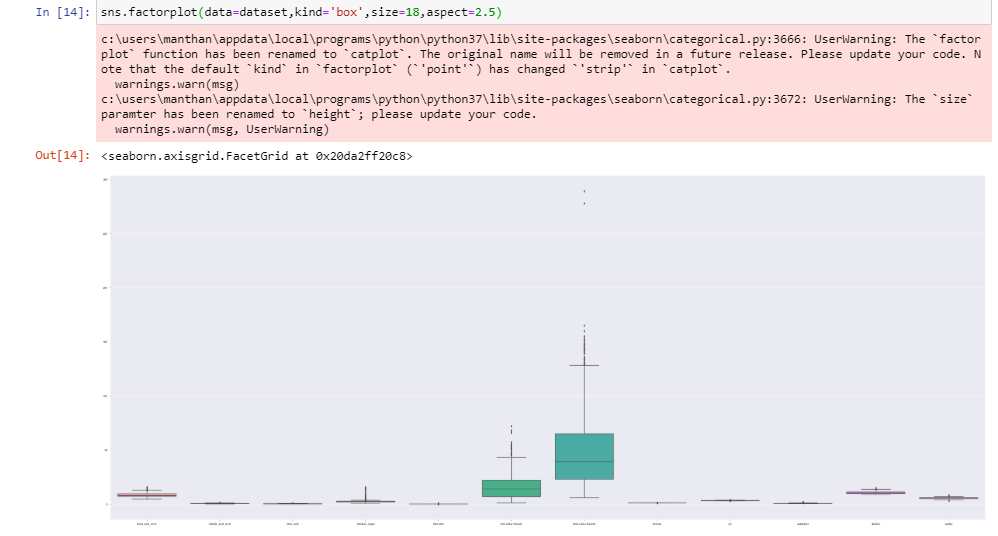




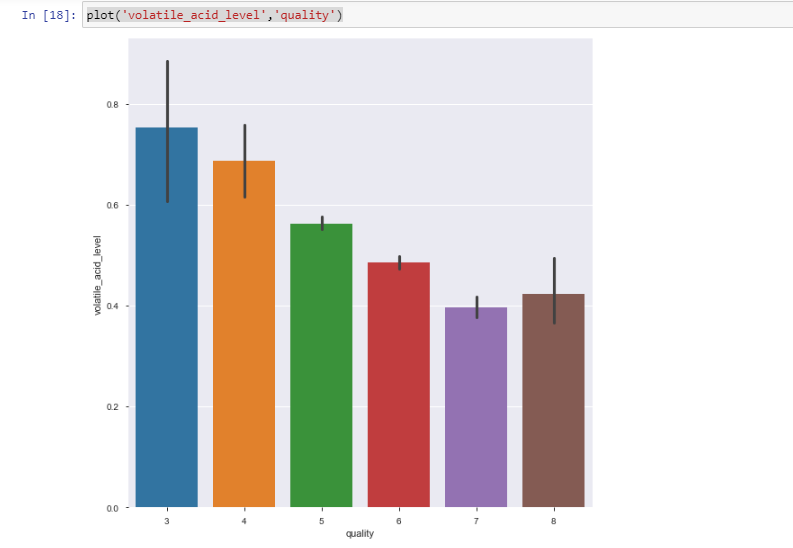
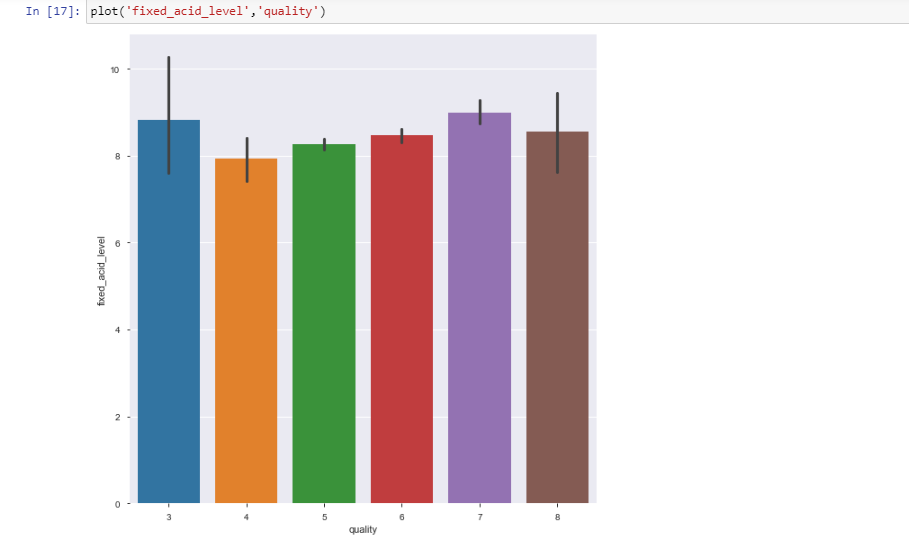
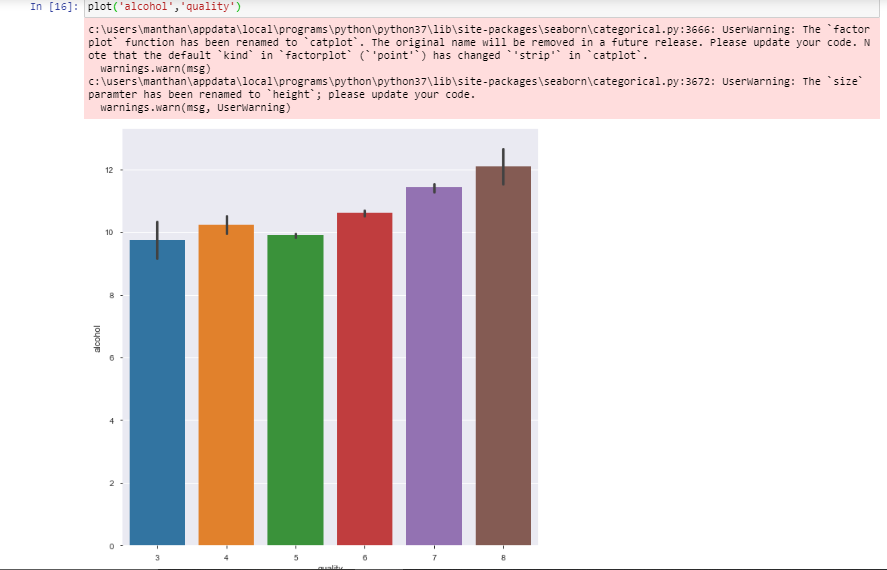
**Data Visualization**

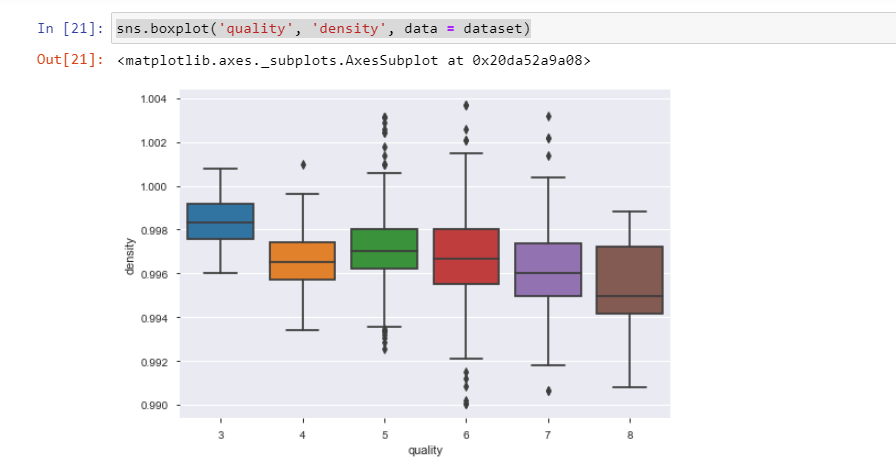
Data visualization can be achieved via the use of bar graphs and box plots.



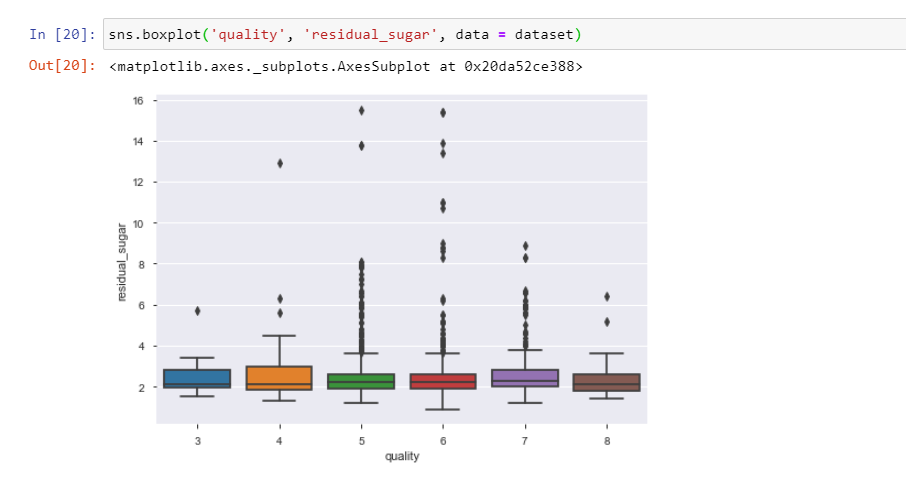


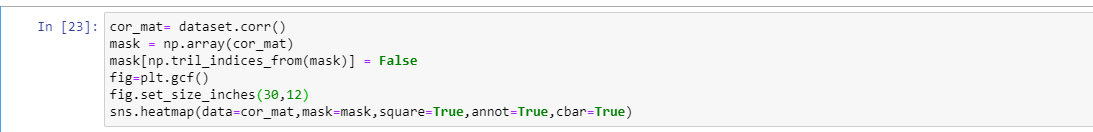


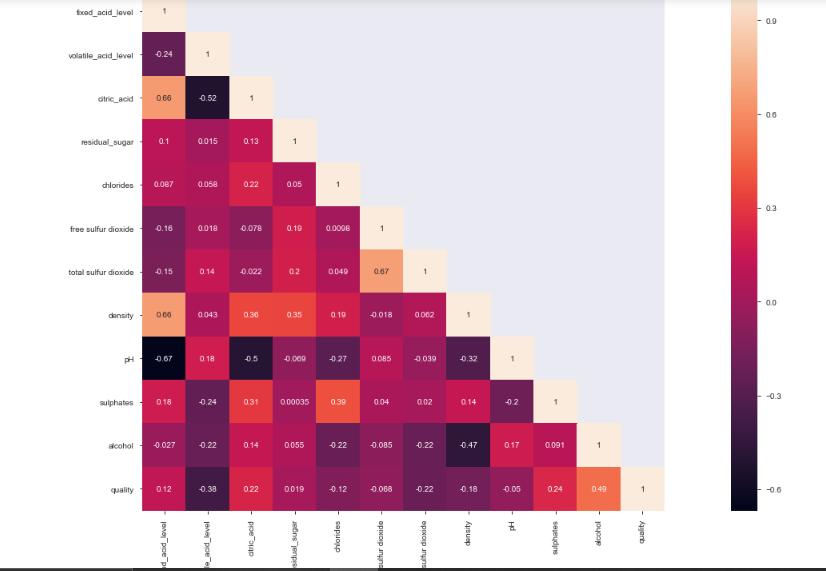






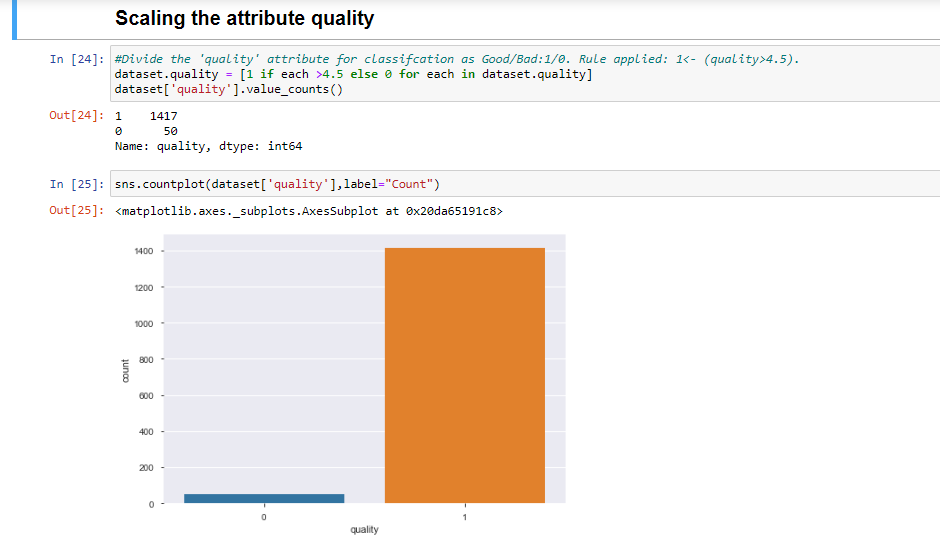






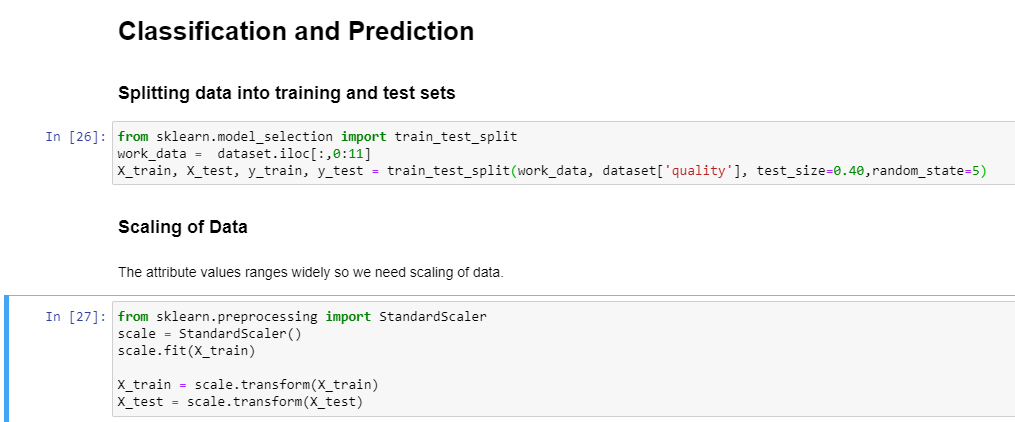
**Scaling the attribute ‘quality’**

The dependent attribute ‘quality’ is discretized. Values greater than 4.5 are mapped to 1, and those lesser are mapped to 0.



**Classification and Prediction**

Before actual prediction, the dataset is divided into a training set and a test set as part of random sampling.



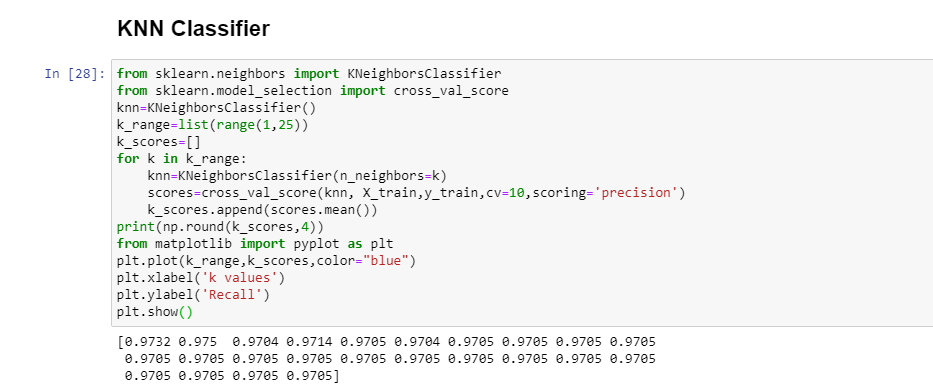
**KNN Classifier**

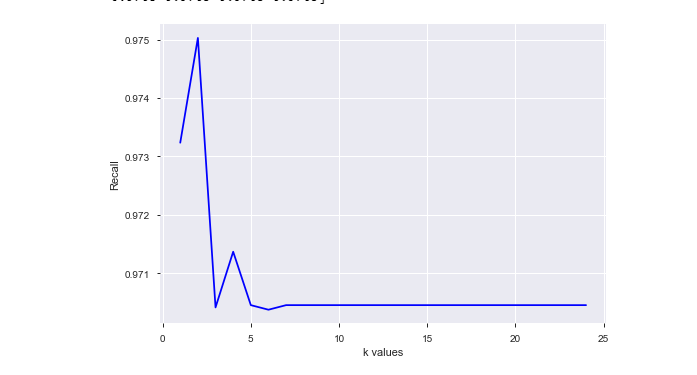
Since the dataset is small and fully numerical, KNN classifier is feasible and make a good choice as a classification model.

KNN classifier from the sklearn library is used.

#### **Choosing the value of k**

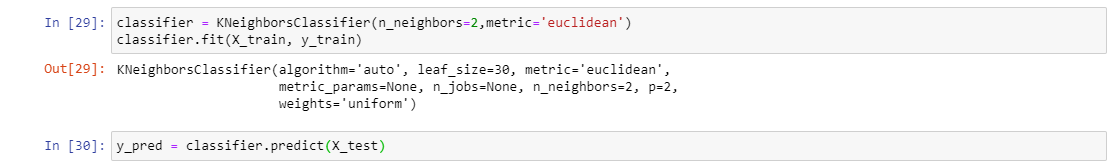
First, we choose a value of k, based on the plot for precision values ranging from [1,25].



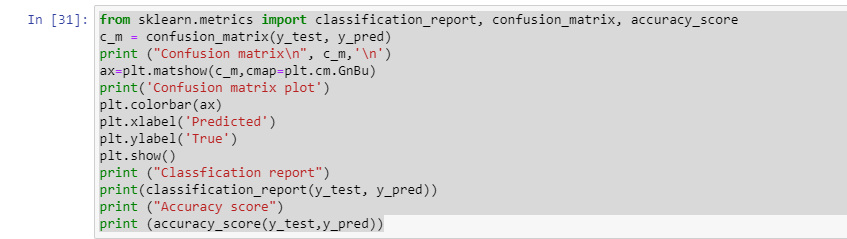


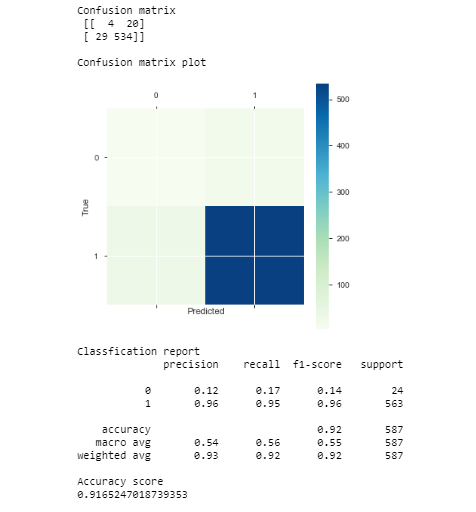
#### **Model training**

Plot of Precision v/s k shows that the best precision lies around k=2. On the basis of this observation, we fix k=2 for classification through the algorithm. For the KNN classifier, we take distance metric as Euclidean distance.



#### **Classification Report**

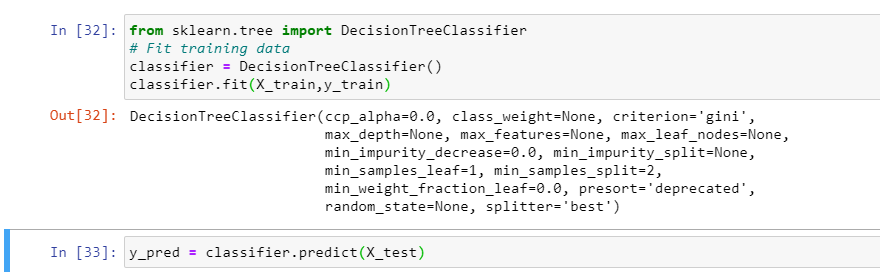




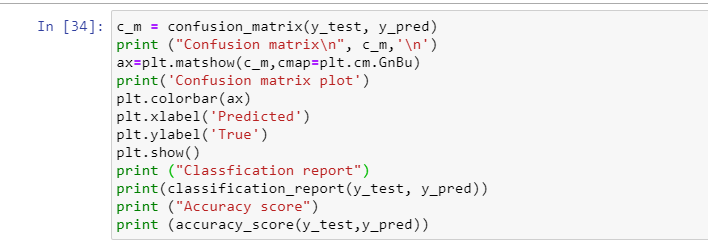
Thus, KNN algorithm gives 91% accuracy for our analysis.

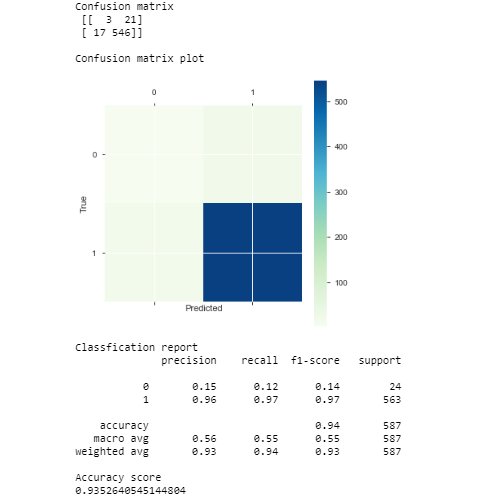
### **Decision Trees**

The Decision Tree classifier is used since attributes are numeric and split point can be easily found on the basis of Gini index as a measure of impurity.



**Classification Report**





The Decision Tree classifier gives accuracy value 93%

**Conclusion**

Both algorithms are predicting high accuracy levels which shows that our analogy is true and both of them can be used for classification of this dataset. Decision trees is giving a better result with accuracy 93.5, whereas the accuracy shown by KNN classifier is 91.65.