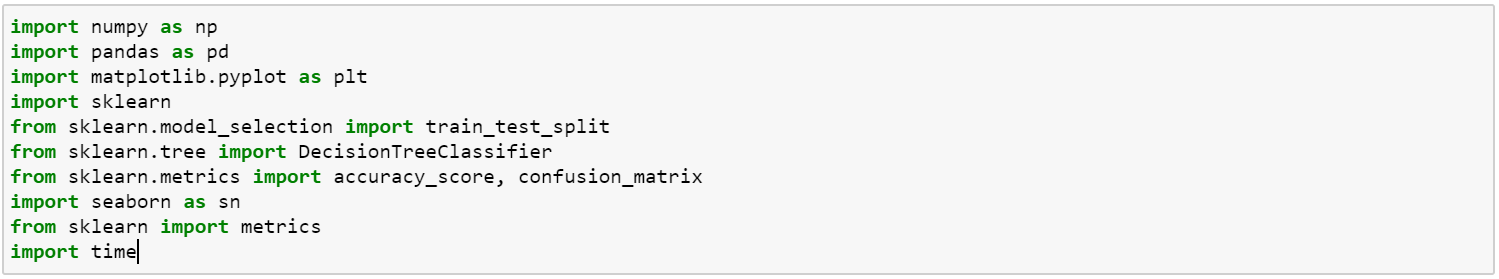
**Decision Tree – (Model-2)**

**Sharing the process**

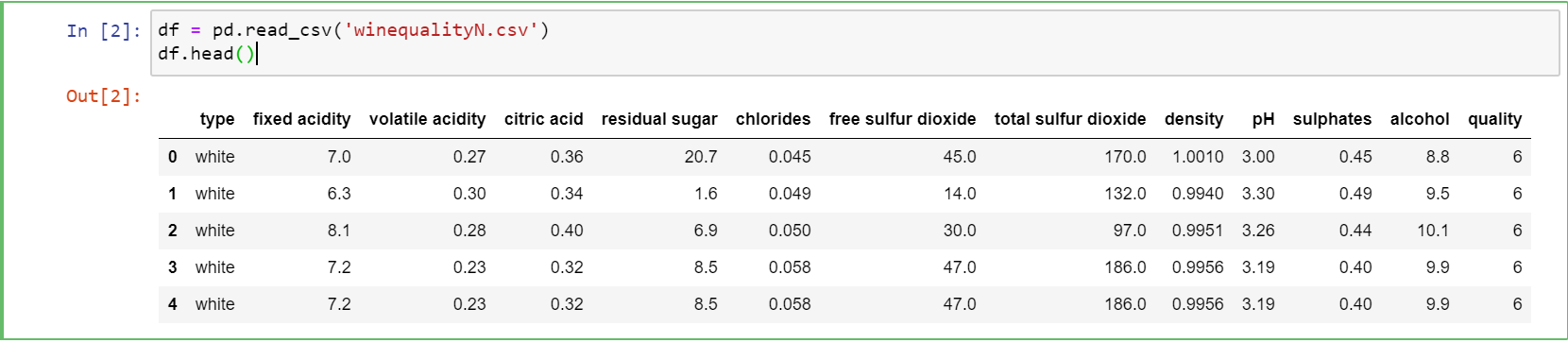
**Without PCA**

**Data before Data Engineering**

Import the necessary libraries such as numpy, pandas.



Use read\_csv() function to read our dataset.



**Data after Data Engineering**

Checking NULL values in data.



Copy the original dataframe to a new dataframe to replace null values with the mean of each null values columns.



The first column ‘Type’ is categorial values called white, red.

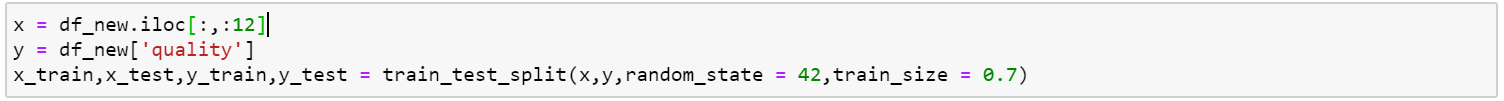
Therefore, we map each type with class 0 and 1.



After that we split training and testing data by using train\_test\_split.

We use all of the columns except quality as our training data.

In this case, we take training size 70% of data and 30% is for testing



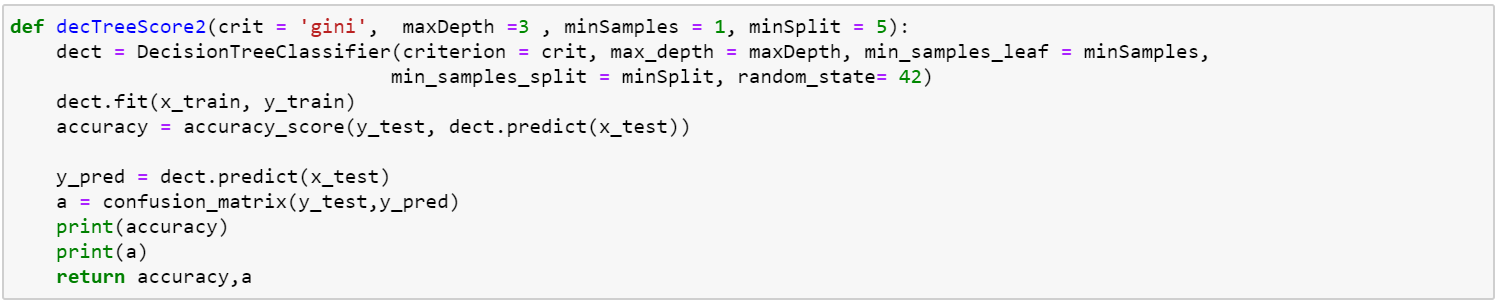
The below function is to feed our training data to the model and get the accuracy results.

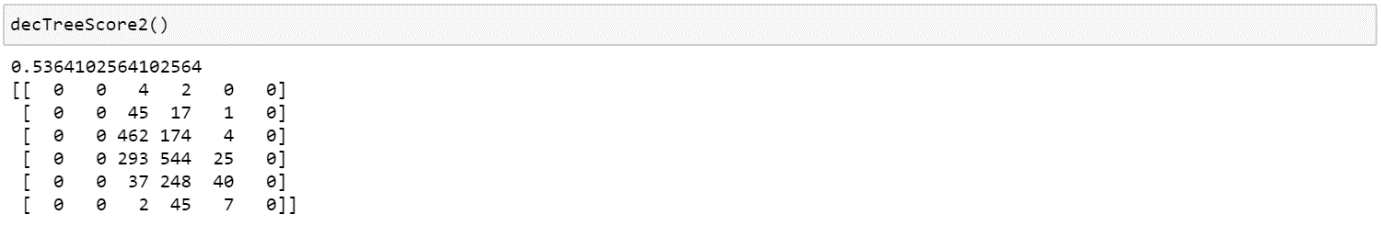
In this decision tree classifier, we use maximun depth = 3 that gives accuracy 0.5364.

Increasing maximun depth will give **better** accuracy results.

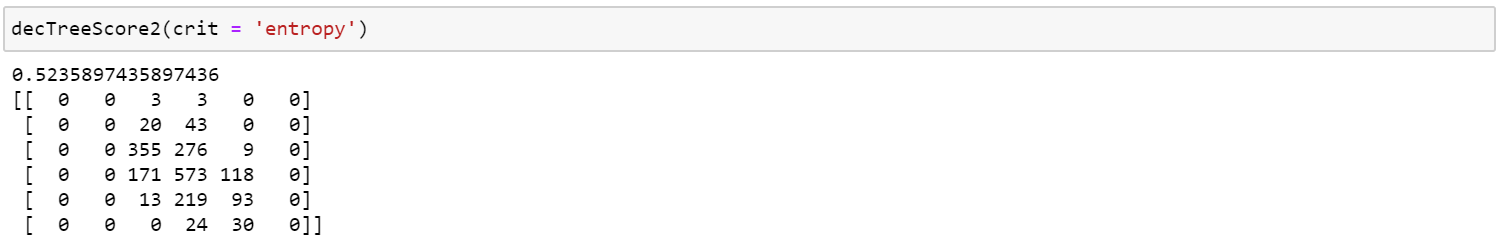
depth = 3 , accuracy = 0.5364

depth = 5, accuracy = 0.5605



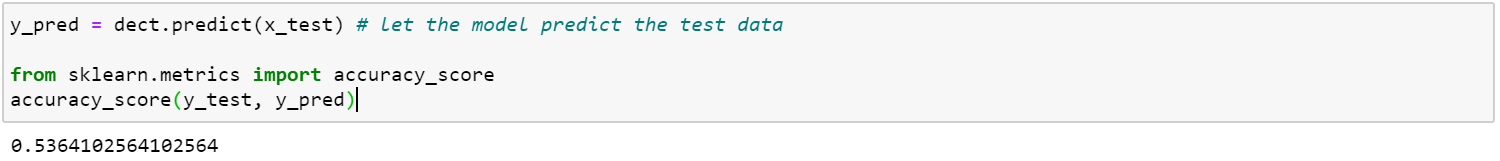


Although criterion ‘entropy’ and ‘gini’ have different calculations, we have very near result accuracy.



This is our prediction for testing data.

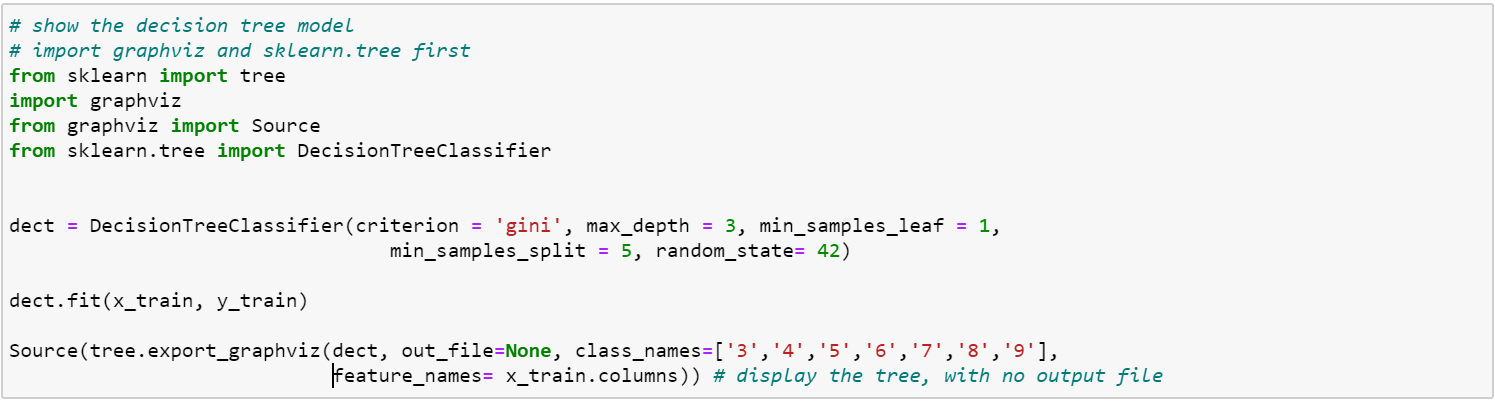
Accuracy is 0.53.

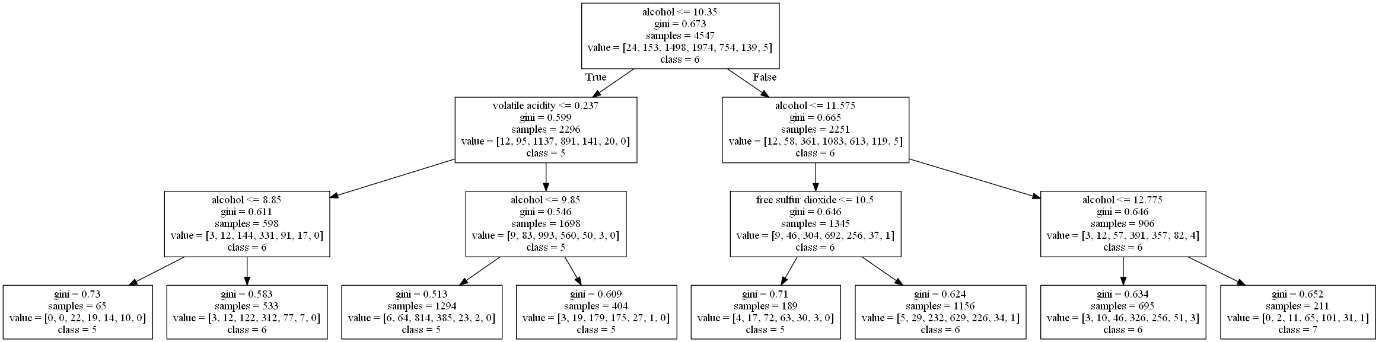


Compare the result of actual values and prediction values.

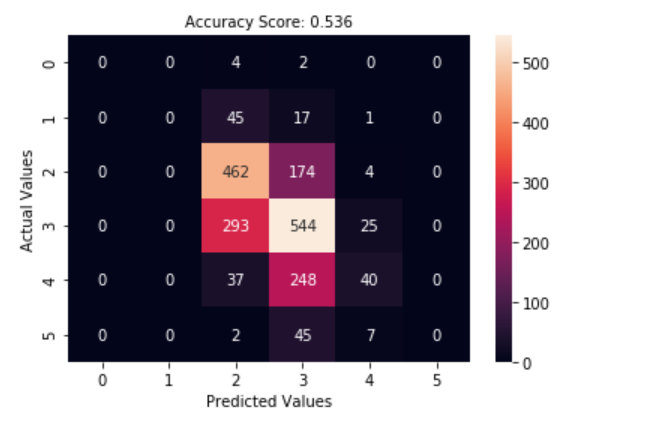


Generating decision tree result with export\_graphviz() function.





A confusion matrix that **describe the performance of a classification model** on a set of test data for which the true values are known.



#Need to add training duration

**In conclusion, the classification accuracy for wine quality is nearly 60%.**

**To get better accuracy results, we will implement new model by using binning method.**

**Decision Tree – (Model-3)**

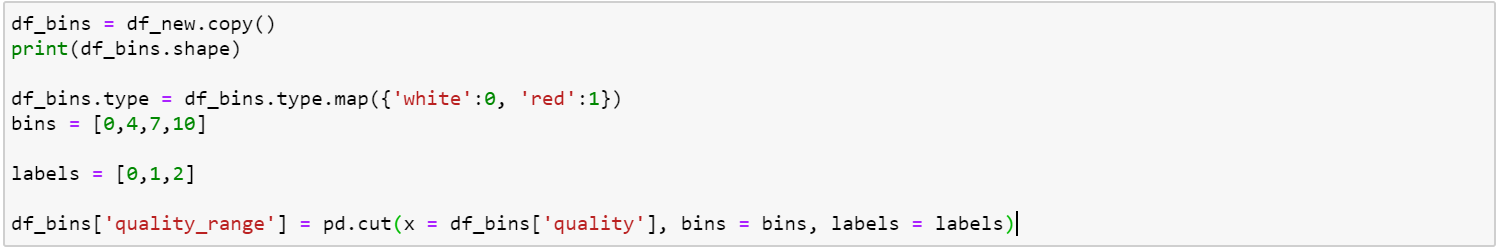
**Without PCA**

Since the accuracy score is not good in model-2, we use **binning method** to get the higher accuracy. We divide the quality column into quality group (0,1,2) and the quality range is

0 – 4 is quality group 1 (Low)

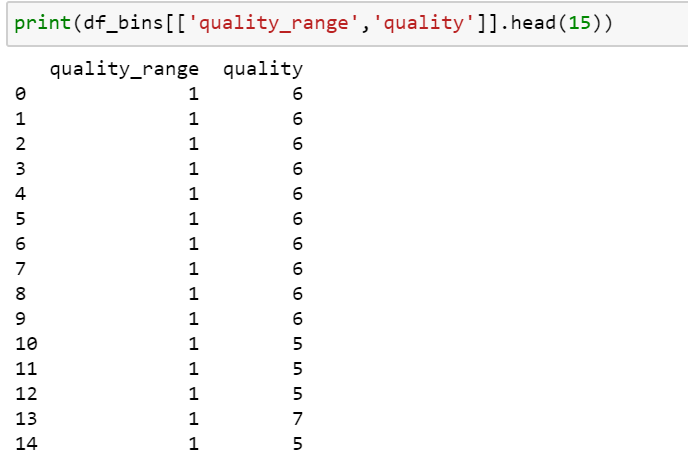
4 -7 is quality group 2 (Median)

7- 10 is quality group 3 (High)



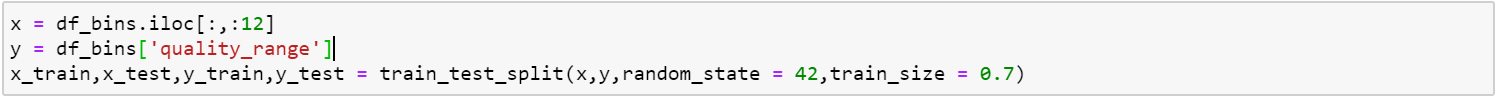
The result of using binning looks like this.

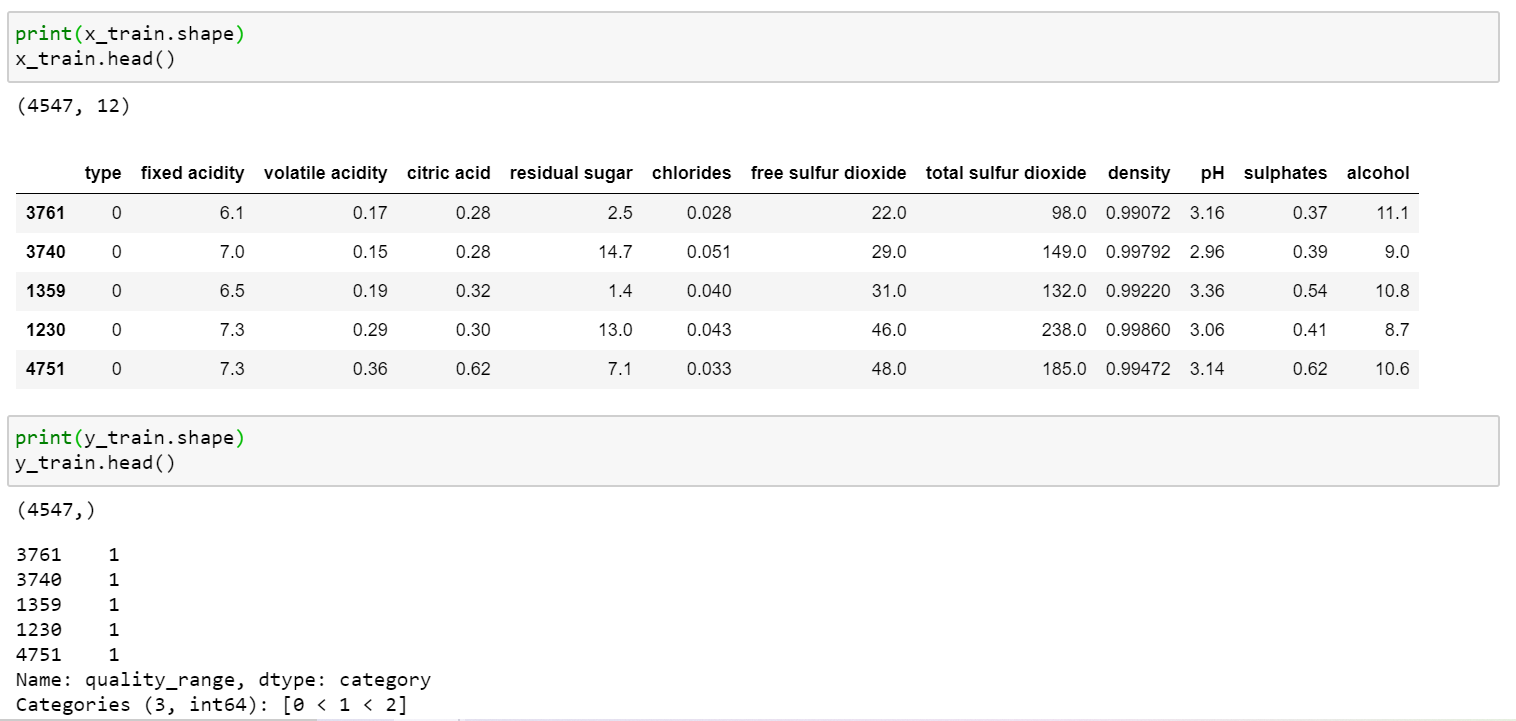
Quality 6 involves in quality range 1.



After that we split training and testing data by using train\_test\_split.

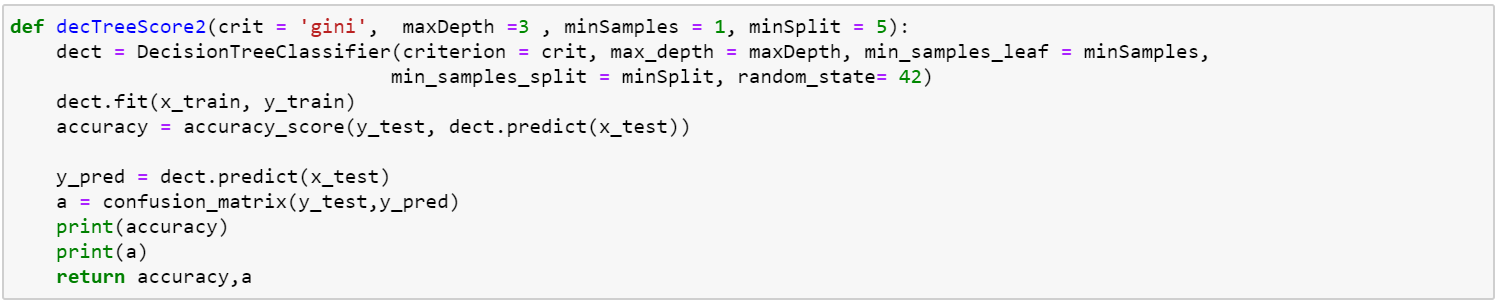
In this case, we take training size 70% of data and 30% is for testing.





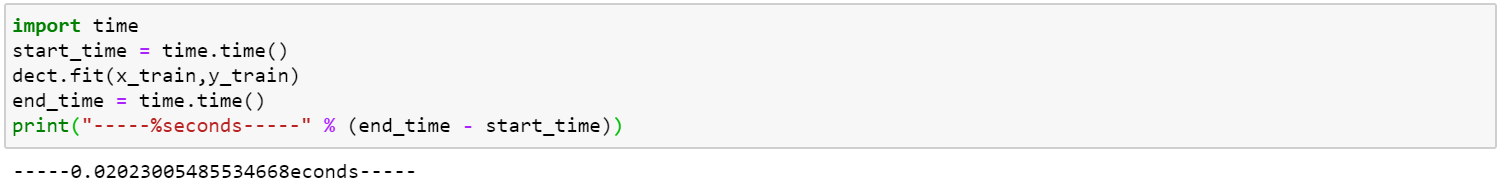
This decTreeScore2 function use maxDepth 3, minSamples = 1 and minSplit = 5 gives the accuracy of 0.93.

We got a better result by comparing the model-2.

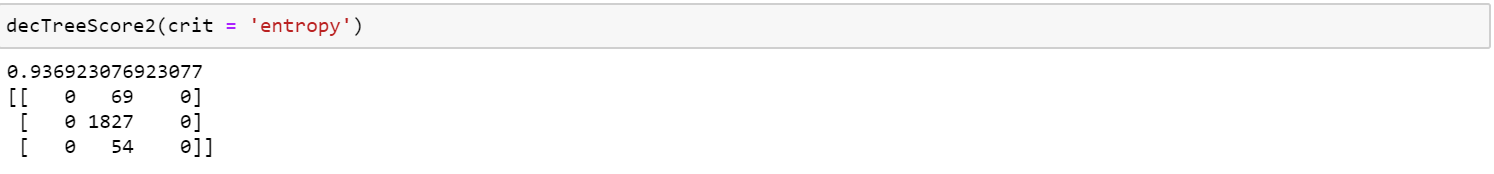


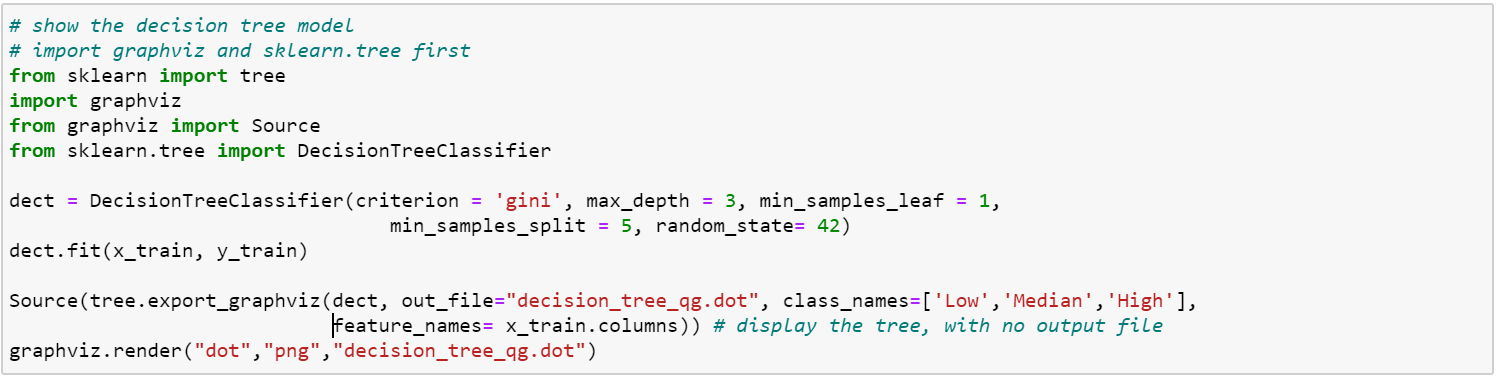


Training Model Duration

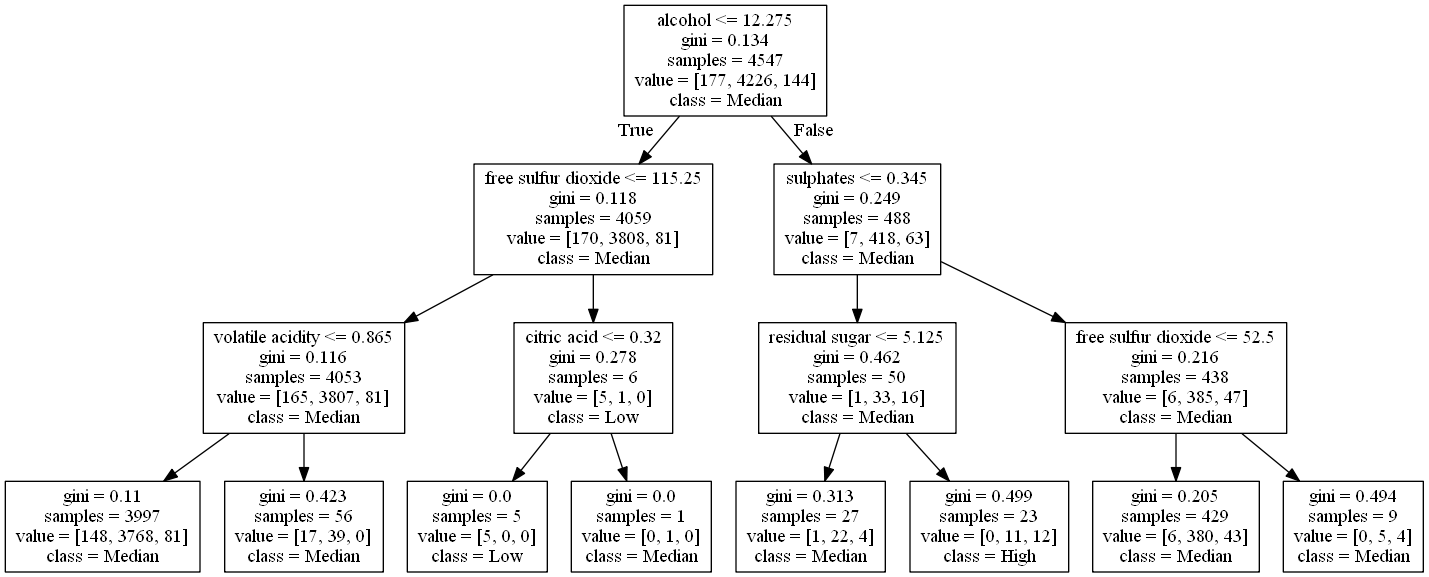


Decision Tree with criterion ‘Entropy’ will give similar accuracy with ‘Gini’

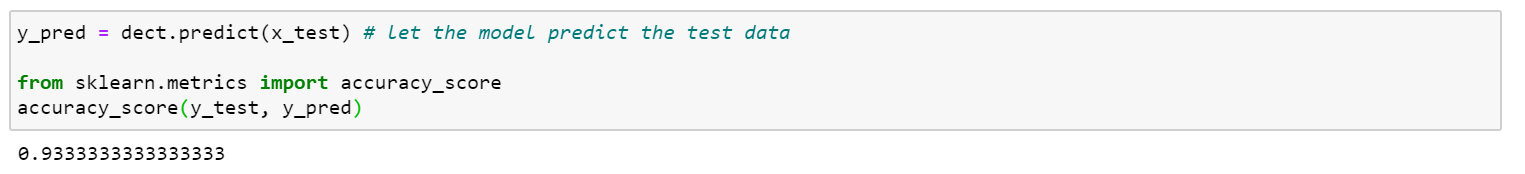




By using export\_graphviz, our decision tree with depth = 3 gives the class results.



Accuracy is 0.93 for testing data.

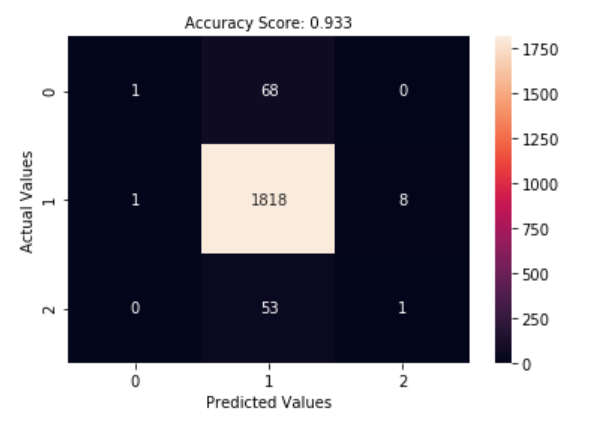


We can also see the actual testing data and prediction results.



This is our confusion metrix that displays the accuracy score of 0.93.





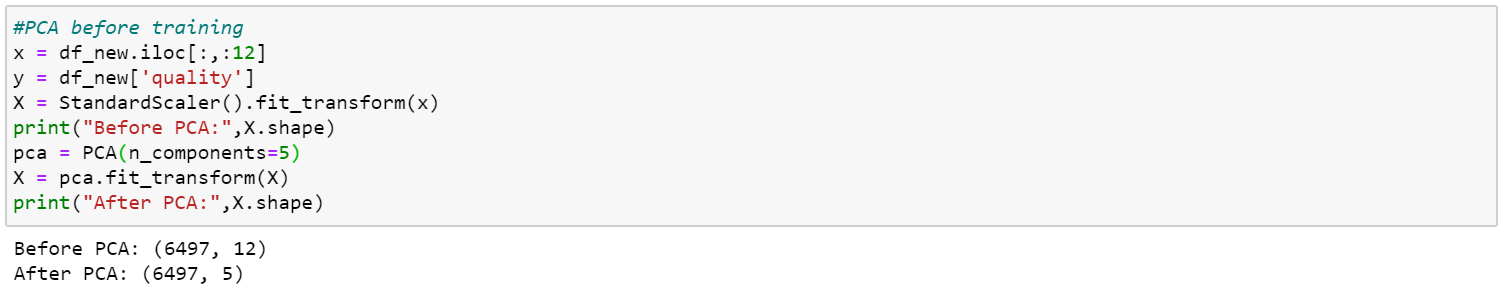
**Feature Engineering**

**Feature Extraction**

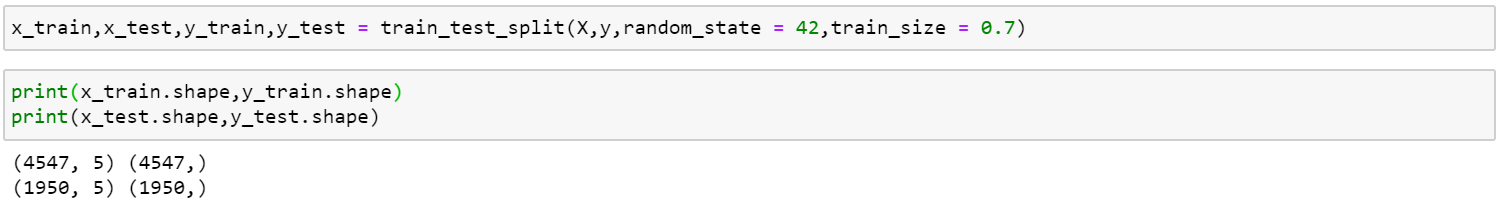
**Model-2 - With PCA**

We use PCA on model-1 to reduce features on our data.

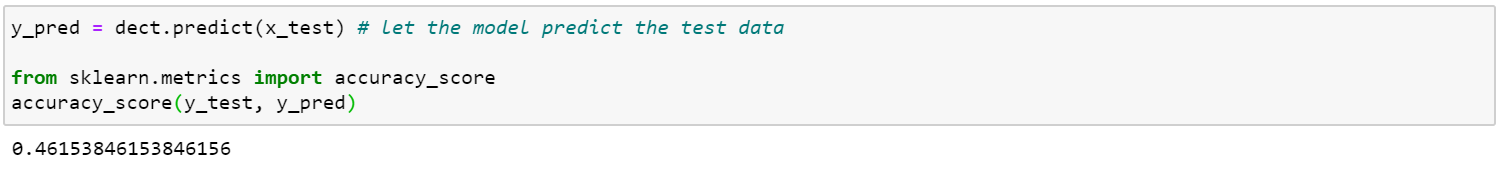
Before PCA, dataset has 12 features on training and 5 features after PCA.



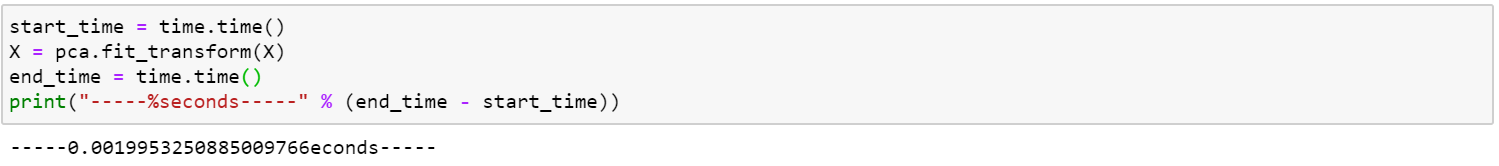
Now we use new X after PCA on train\_test\_split to get training and testing data based on this PCA features.



Although PCA is applied to model\_1, we don’t have better accuracy result.



Duration of training on PCA

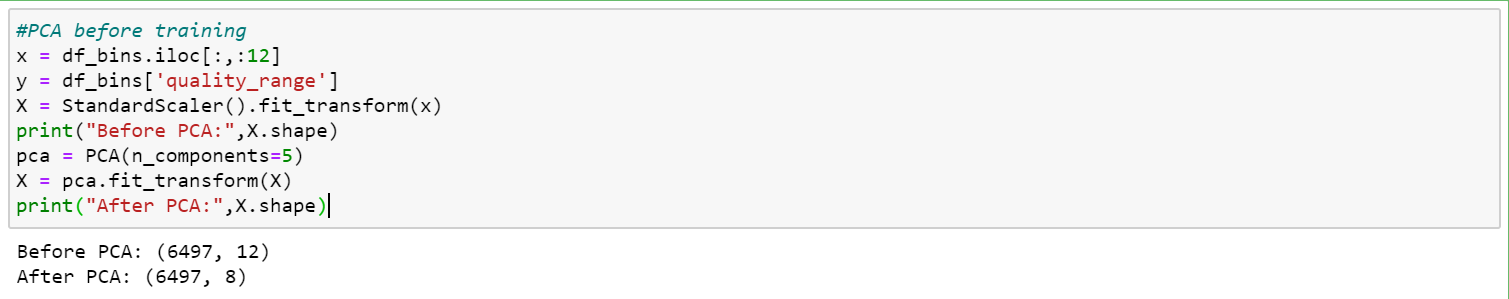


**Feature Selection**

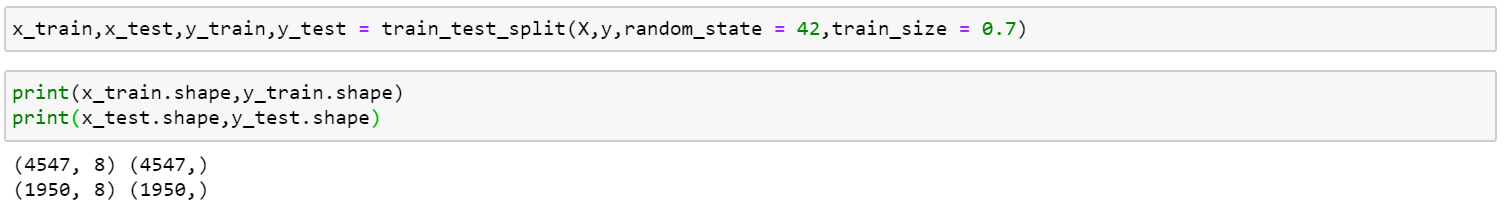
**Model-3 – With PCA**

We also use PCA on model 3 that use binning method in our model to reduce features.

Before PCA, 12 features on training dataset and get 5 features after PCA.



Now we use new X after PCA on train\_test\_split to get training and testing data based on this PCA features.

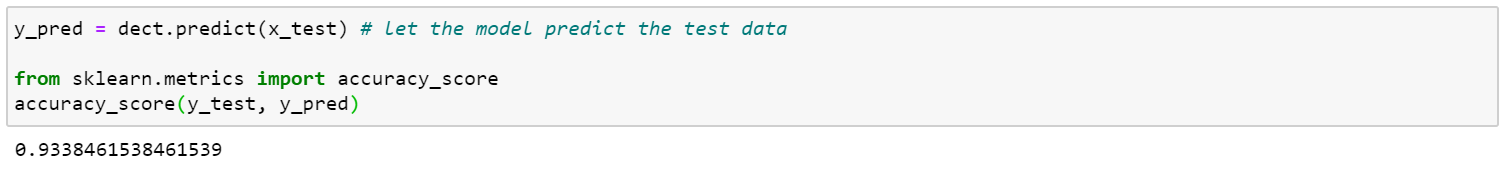


Without PCA with 12 features has accuracy of 0.9333.

With PCA with 5 features has accuracy of 0.93384.

Although we train our model with 5 features instead of 12 features, we can get the same accuracy result.

In this case, the more components we have, the better accuracy we will have.



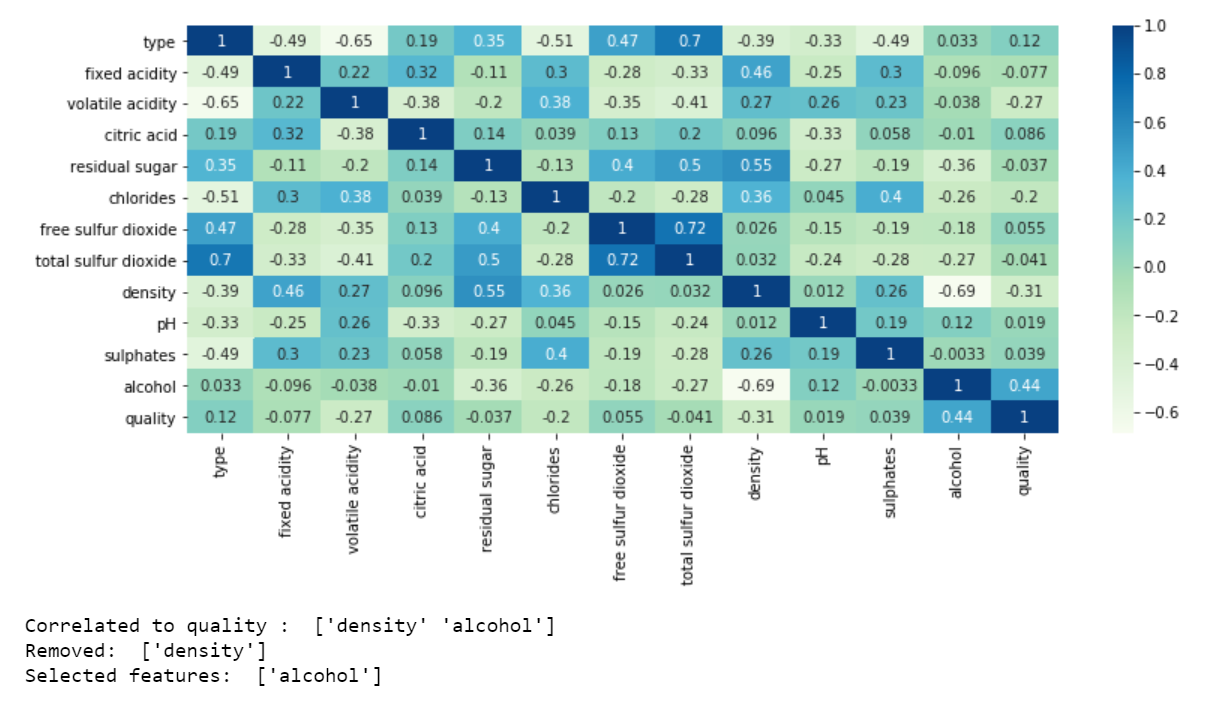
#Duration of training on model-3(quality group) with PCA.



**Feature Selection by Pearson Correlation Matrix**

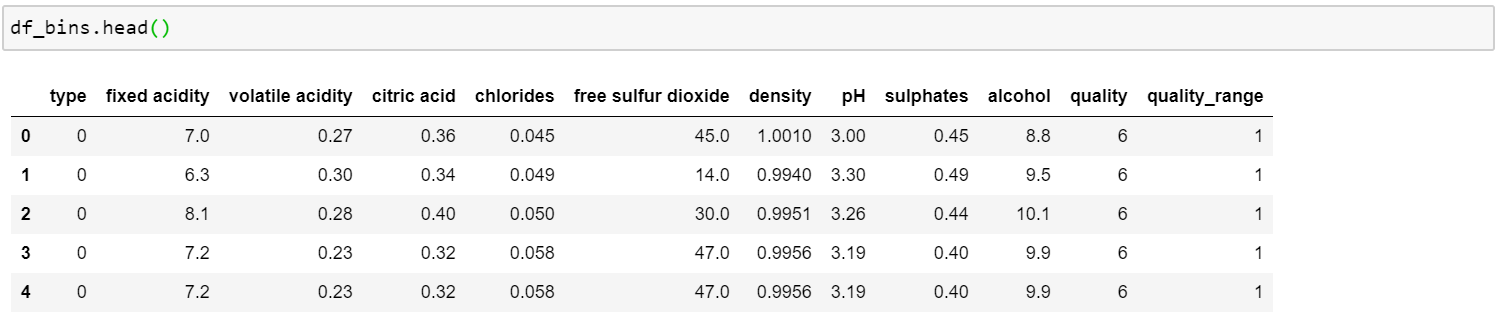
By Using Pearson Correlation, we can know

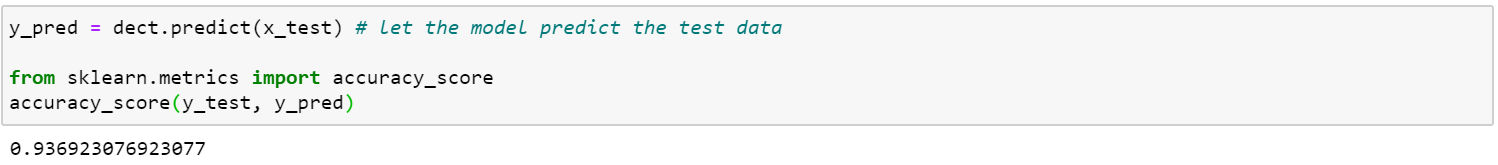
* ‘quality’ is correlated with alcohol.
* ‘Free sulfur dioxide’ and ‘total sulfur dioxide’ is highly correlated (0.72 in our confusion matrix) and they can have almost the same effect on an independent variable.
* Therefore, when the two features are highly correlated, we can drop one of them in our feature set.
* Correlation of ‘total sulfur dioxide’ and ‘residual sugar’ is 0.5 and they have moderate correlation.
* Correlation of ‘density’ and ‘residual sugar’ is 0.55 and they also have moderate correlation.
* In conclusion, our target variable is correlated with alcohol and low correlation with other features.



Dropping both ‘total sulfur dioxide’ and ‘residual sugar’ (or) dropping one of them gives better accuracy result on quality range if we compare to without feature selection.





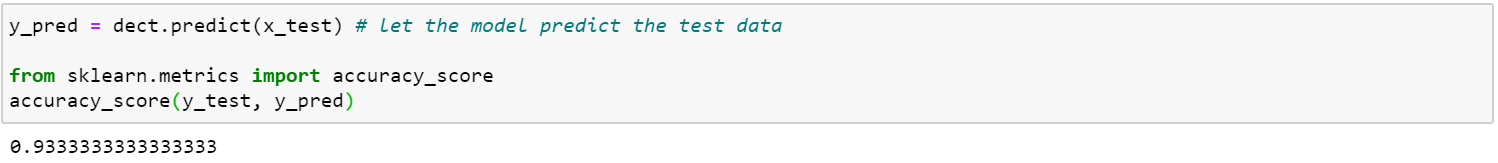


We can also drop one of the features from ‘free sulfur dioxide’ and ‘total sulfur dioxide’ since they are highly correlated.

The accuracy that results after dropping one of them makes no difference in our models.

As mentioned below, the accuracy is 0.93333.





**Impact of Data Engineering and Feature Engineering**

In our dataset, We use two types of data engineering.

* The first one is categorical data engineering. By using categorial data engineering, we map ‘type’ category to a unique value such as white = 0 and red = 1. By using label encoding, we can easily visualize the relationship between the features and the target in our plots and matrix.
* The second one is data cleansing. Since our dataset contains missing data and NaN, we fill with a value of mean value on NaN column.

For the feature engineering, we use feature selection and feature extraction techniques.

* By using Pearson correlation matrix, we can know which features are correlated each other.
* Since ‘free sulfur dioxide’ and ‘total sulfur dioxide’ are highly correlated with each other, removing of these features doesn’t effect on our accuracy results.
* Moreover, dropping ‘sugar’ or ‘total sulfur dioxide’ will give better accuracy results since these are loosely correlated with our dependent variable ‘quality’.