1. **Red Wine Quality**
   1. **Introduction**

There are 11 input variable columns and 1 output variable column in total. These are the eleven input variables:

Fixed Acidity - The sample's fixed acidity is defined as the low volatility organic acids.

When acetic acid levels in wine are high, it can give the beverage a sour, vinegar-like flavor. This is known as volatile acidity.

Citric Acid: It gives wines more flavor and freshness. A weak organic acid without color, citric acid is.

The quantity of sugar left in the wine after fermentation has stopped is known as residual sugar. The sweetness of the wine is influenced by the residual sugar content.

Chlorides: This term refers to the wine's salt content. The ions are what cause the high level of chloride produced while creating red wine.

Free sulfur dioxide - made up of both the SO2 that is present in the wine free-floating and the SO2 that is bonded to other substances like aldehydes, pigments, or sugar.

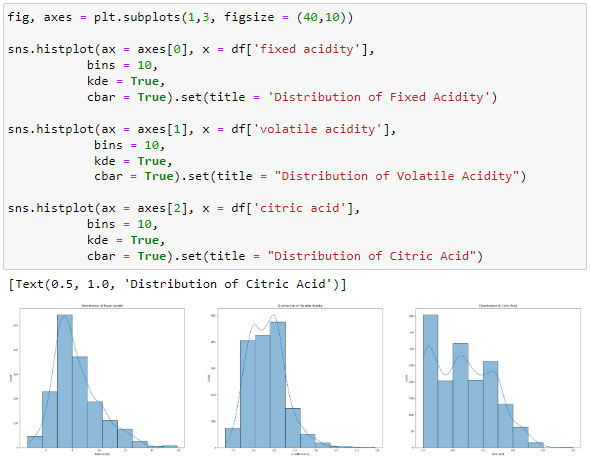
Density - A hydrometer, which is a sealed glass tube, is used by winemakers to gauge the density of juice, wine in the fermentation process, and finished wine in comparison to pure water.

pH - On a scale of 0 to 14, pH indicates how basic or acidic a wine is. The most basic element is 14, while O is the most acidic.

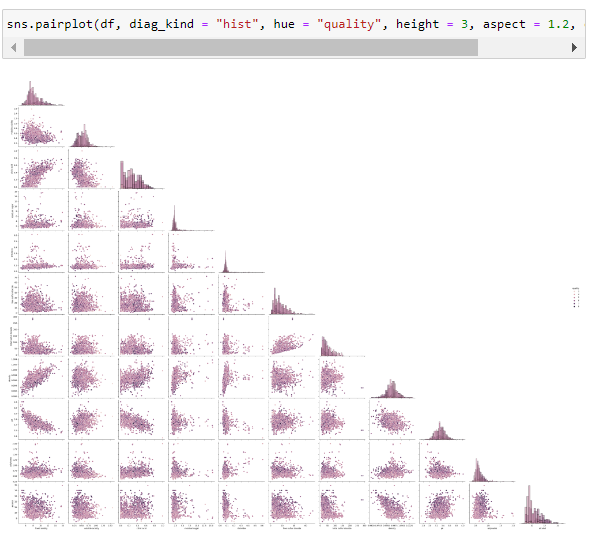
Sulphates - Sulphates are a wine ingredient that raises the level of sulfur dioxide (SO2), which has antibacterial properties. In all wines, it present naturally in little amounts. To preserve and shield the wine from microbial and yeast contamination, winemakers also added sulfites.

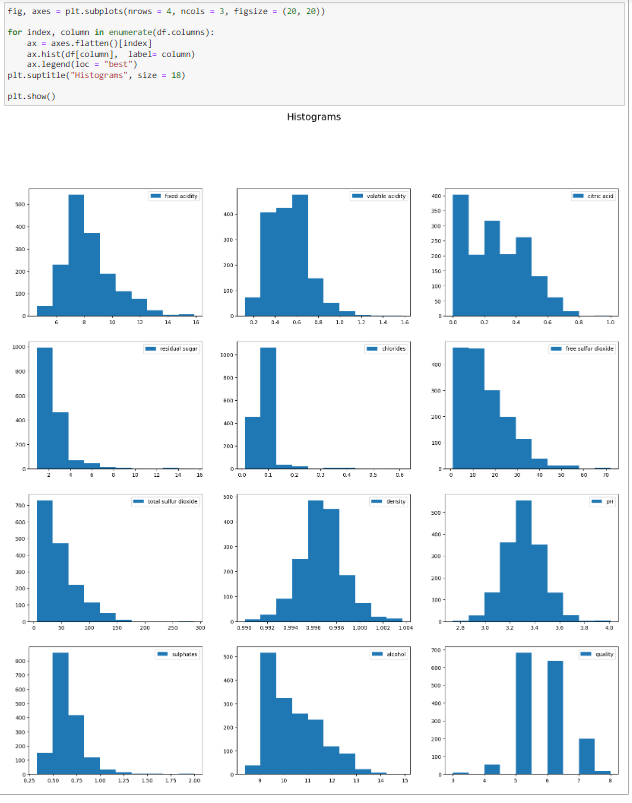
Alcohol - The wine's alcohol concentration as a percentage.

The statistics provides us with the average, standard deviation, minimum, 25th, 50th, 75th, and maximum values for each variable.



Distribution: It turns out that the constant acidity variable's values are largely normally distributed. However, the distribution of the other "volatile acidity" and "citric acid" variables has two peaks.­­



We can see that several factors are correlated. 'Multicollinearity' is the name of the issue. Multicollinearity emerges because independent variables should be independent, this association is problematic. When fitting the model and interpreting the findings, it may be difficult if there is a high enough degree of correlation between the variables.

What do we see when we observe the histograms above? Here we see the distributions of values of all variables. As it can be seen from the charts, the values of 'pH' and "density" variables are relatively normally distributed.

Most of the values of the "fixed\_acidity" variable are in the range of 7 - 8

Most of the values of the "volatile\_acidity" variable are in the range of 0.4 - 0.7

Most values of the "citric\_acid" variable are in the range of 0.0 - 0.1

Most of the values of the "residual\_sugar" variable are in the range of 1 - 2.5

Most of the values of the "chlorides" variable are in the range of 0.085 - 0.15

Most values of the "free\_sulfur\_dioxide" variable are in the range 0 - 15

Most values of the "total\_sulfur\_dioxide" variable are in the range 0 - 30

Most of the values of the "density" variable are in the range of 0.996 - 0.998

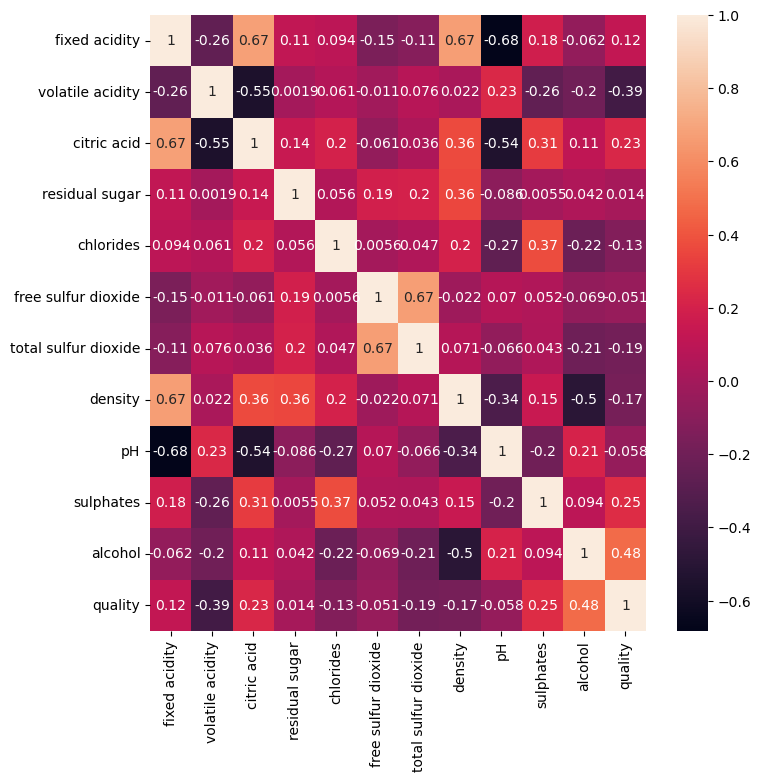
Most of the values of the "pH" variable are in the range of 3.2 - 3.4

Most of the values of the "sulphates" variable are in the range of 0.50 - 0.75

Most of the values of the "alcohol" variable are in the range of 9 - 10

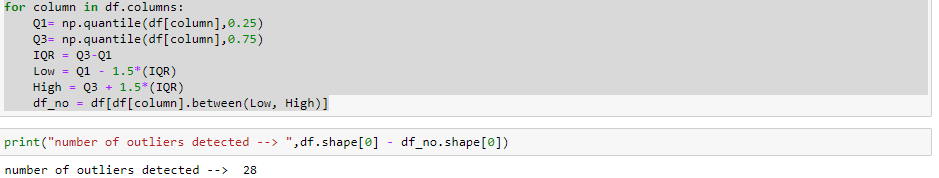
Most values of the "quality" variable are 5 and 6.

**Heatmap**



We can observe that the variables "free sulfur dioxide" and "total\_sulfur\_dioxide" have a relatively strong (0.67, positive) association. The "pH" and "fixed\_acidity" variables have a negative association that is comparatively strong (-0.68). The multicollinearity issue is brought on by the significant correlations between alcohol and density (-0.49), citric acid and pH (-0.54), and free and total sulfur dioxide (-0.67).

* 1. **Handling Outlier**

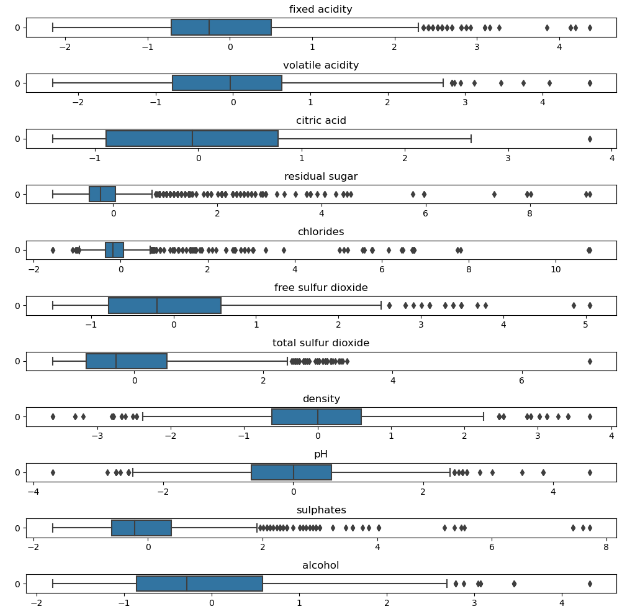
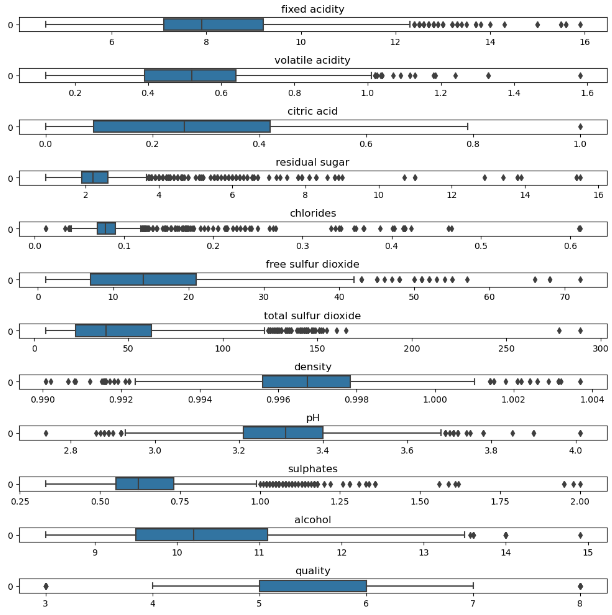


28 rows of outlier is detected and dropped from the column so that extreme values doesn't affect our machine learning models.

We can see that there are still some outliers on the columns but I have decided to keep those outliers to prevent data from overfitting.

Most of the dropped data are from the outliers from residual sugar column.

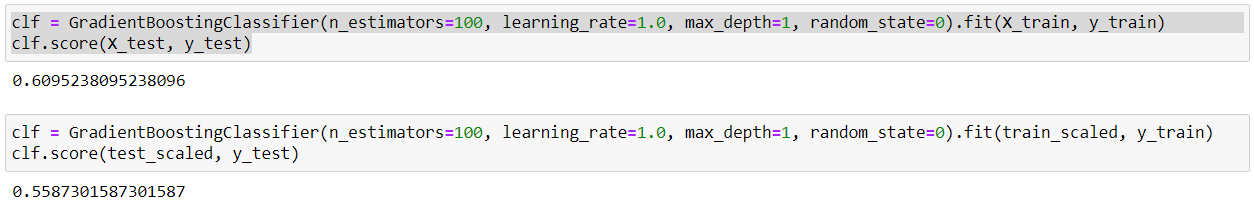
* 1. **Scaling the data**



Unscaled Data Scaled Data

Scaling helps to bring the features onto a similar scale, ensuring that no particular feature dominates the learning algorithm simply because of its larger magnitude. Features with higher magnitudes might otherwise have a disproportionate impact on the learning process. By scaling the features, you ensure that each feature contributes equally to the model training.

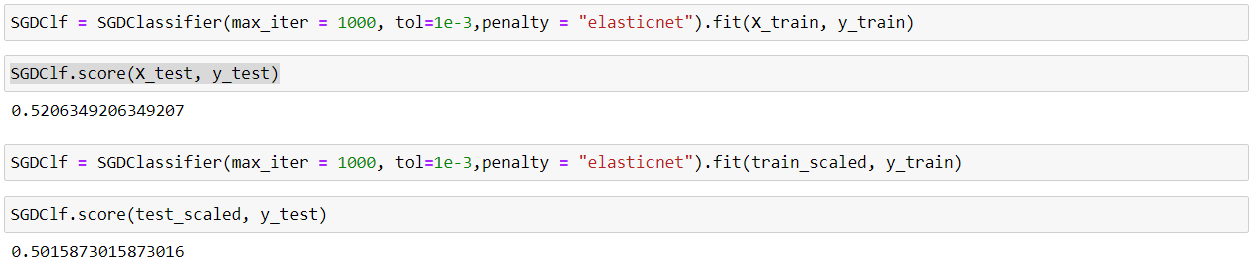
* + 1. **Comparison Between Models with Scaled and Unscaled Data**

The family of ensemble methods includes the well-known machine learning algorithm gradient boosting classifier. It is a form of boosting algorithm that combines several decision trees or other weak or base learners to produce a powerful predictive model.

It shows here that the result for the unscaled data is higher than the scaled data when we use the Gradient Boosting Classifier.

**SGD Classifier**

The linear classifier family of machine learning algorithms includes the SGDClassifier. Problems involving binary and multiclass classification can be resolved by the SGDClassifier.

Due to its stochastic nature, the SGDClassifier is effective and able to handle big datasets. When working with high-dimensional data or in online learning scenarios where fresh data is constantly arriving, it might be especially helpful.

Under the SGD Classifier, the score for unscaled data is higher than scaled data and is lower than Gradient boosting Classifier.

**Support Vector Classifier**

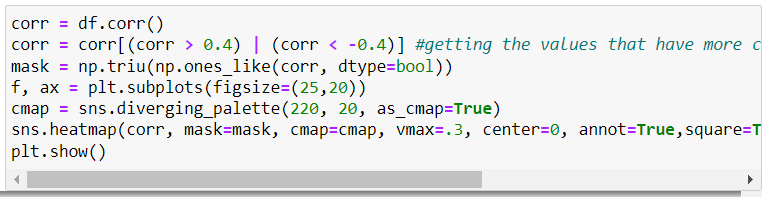
A machine learning algorithm called Support Vector Classifier is employed for classification jobs.

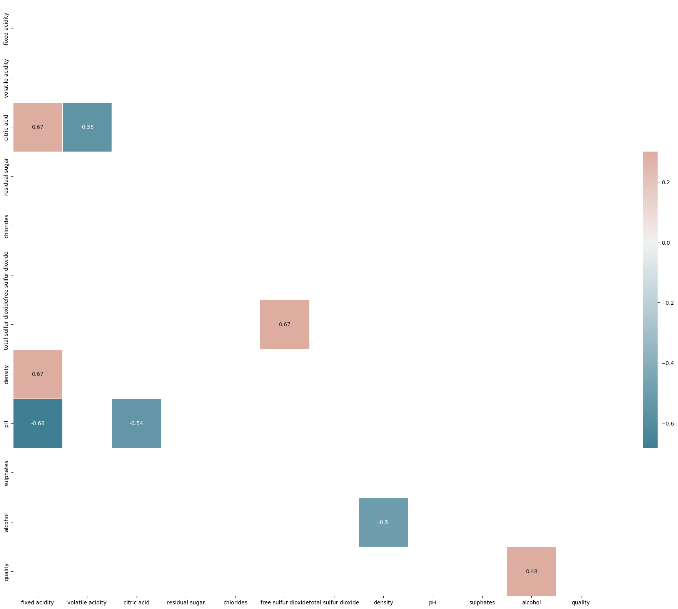
SVC is a potent method for classification jobs that has seen extensive use across several industries. However, it can be computationally expensive for large datasets, and tweaking the hyperparameters is essential for achieving the best performance, such as selecting the right kernel and regularization parameter.



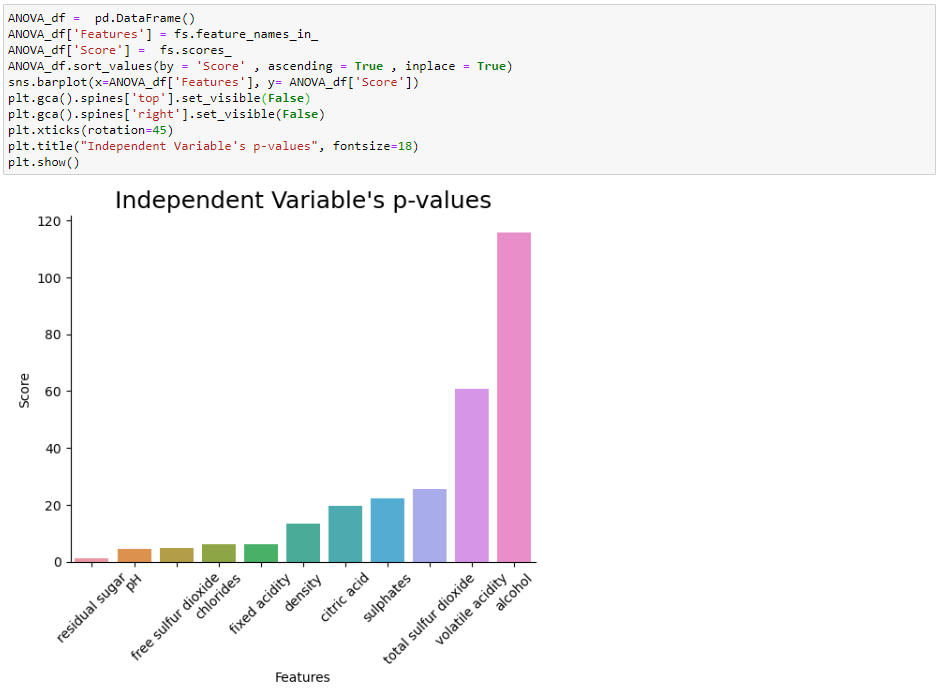
The SVC model scored 0.628 which is the highest score amongst the model.

* 1. **Feature Selection**

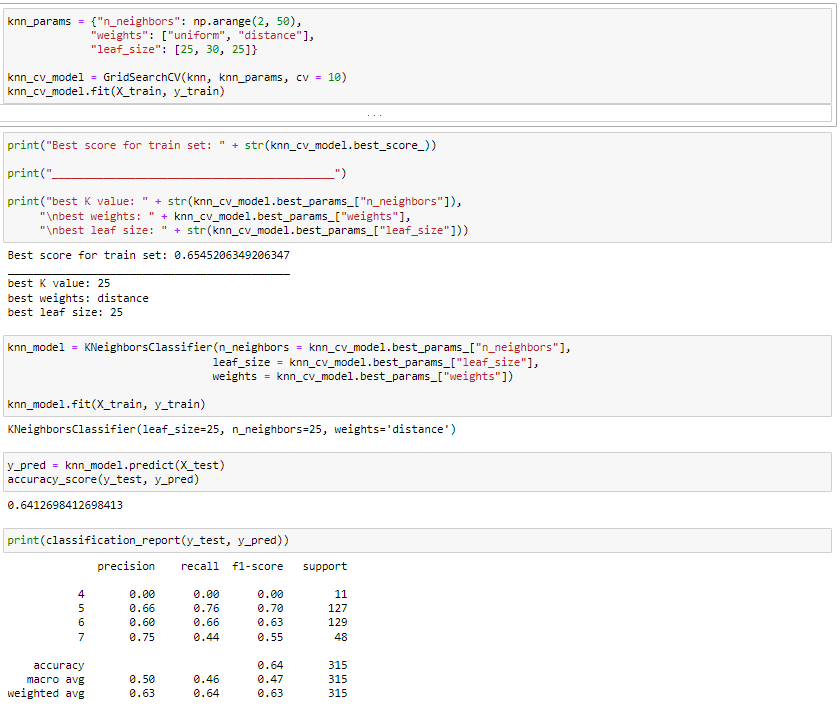


****

Bar plot to visualize the p-values of independent variables.

P-value is a measure that helps determine the statistical significance of a hypothesis test. It quantifies the strength of evidence against the null hypothesis.

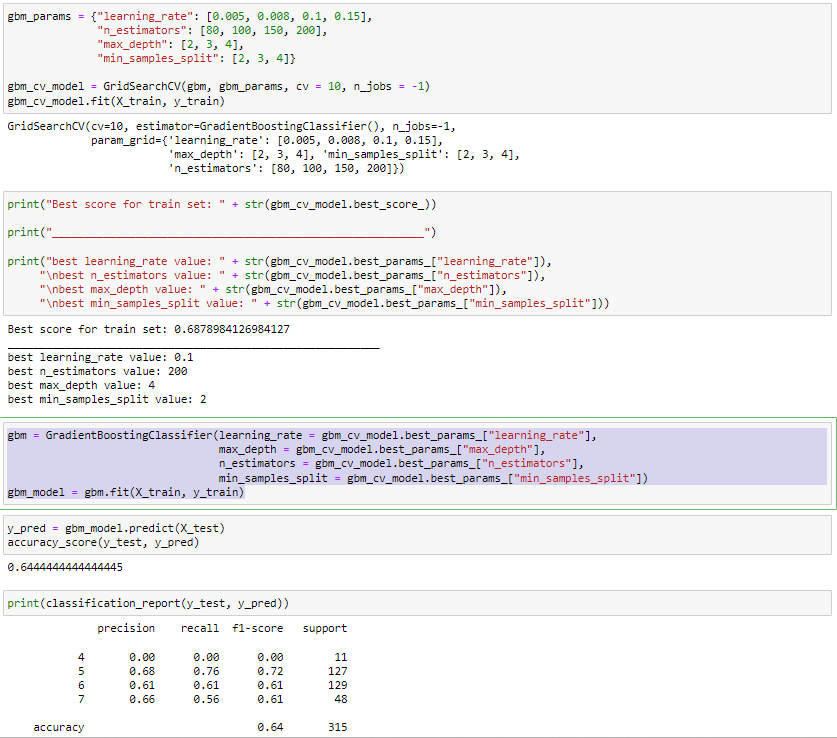
* 1. **K-Nearest Neighbors Algorithm**

A non-parametric supervised learning technique for classification and regression is the K-nearest Neighbors algorithm (k-NN). A class membership is the result. By allocating an item to the class that is most popular among its k closest neighbors, a datapoint is classed by a majority vote of its neighbors. Since the algorithm uses distance to classify data and the characteristics have such a wide range of scales, normalizing the data can significantly increase classification accuracy.

By methodically experimenting with various combinations using cross-validation, the algorithm assists in determining the KNN classifier's ideal hyperparameters, thereby enhancing the model's performance.

The model's total accuracy is 0.58. Performance of the model varies between courses. Poor performance is shown for class 4, mediocre performance is shown for classes 5 and 6, and average performance is shown for class 7. The F1-score gives a balanced measure of precision and recall, while the precision metric assesses the precision of positive predictions and recall assesses the capacity to recognize positive samples.

* 1. **Gradient Boosting Machines(GBM) Model**

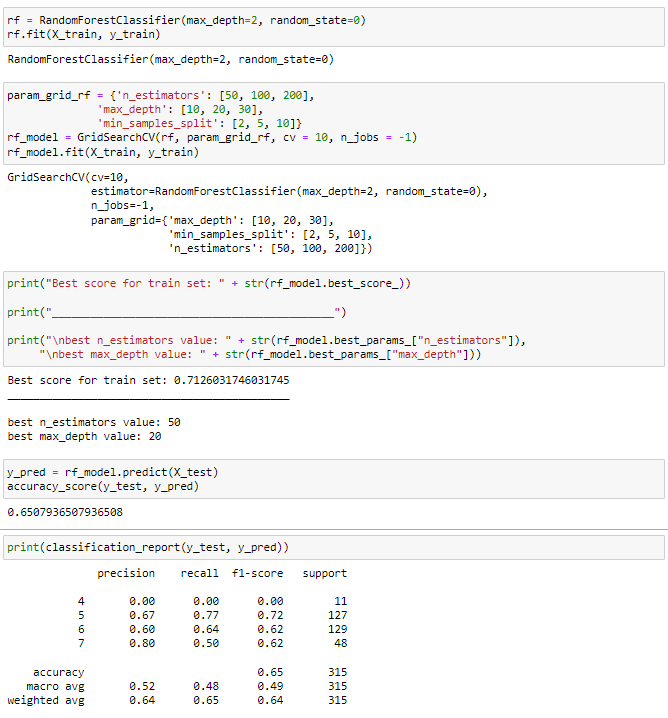
Gradient Boosting is utilized for regression and classification applications. It provides a prediction model in the form of an ensemble of decision trees-like prediction models.

Using the best hyperparameters discovered from a prior grid search, a Gradient Boosting Classifier (GBM) is initialized and trained.

The model's overall accuracy is 0.67, and its performance varies depending on the class.

Performance of the model varies between courses. Class 4 performs rather poorly, while classes 5, 6, and 7 have higher recall rates and F1-scores, indicating a performance that is reasonably balanced between precision and recall.

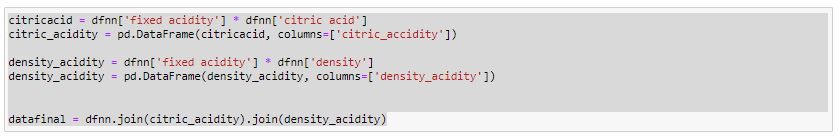
* 1. **Random Forest Classifier**

An technique called random forest combines the results of various decision trees to get a single outcome. Its widespread use is motivated by its adaptability and usability because it can solve classification and regression issues. In the random forest technique, numerous decision trees combine to produce an ensemble that forecasts outcomes more accurately, especially when the individual trees are uncorrelated.

The model's overall accuracy is 0.65. The model performs differently for various classes, with some classes having more accuracy than others. Class 4 performs poorly and receives a recall and F1-score of 0.00. While the other classes in classes 5, 6, and 7 receive respectable marks of 0.72, 0.62, and 0.62.

Class 5, Class 6, and Class 7 have good precision of 0.67, 0.6, and 0.80 respectively, which is good for the model.

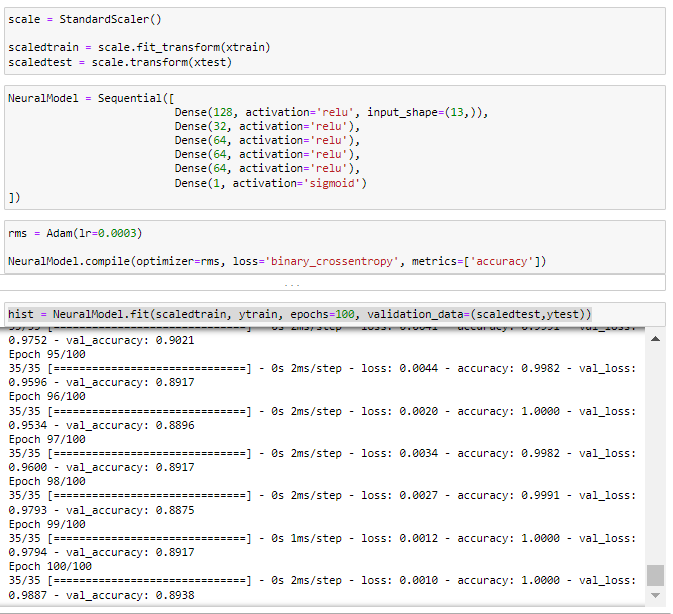
* 1. **Neural Network**

****

Two new columns: 'citric\_acidity' and 'density\_acidity'. The 'citric\_acidity' column is calculated by multiplying the 'fixed acidity' column with the 'citric acid' column in dfnn. The 'density\_acidity' column is calculated by multiplying the 'fixed acidity' column with the 'density' column in dfnn.

****

The Quality label Y is categorizes them into bins defined by **bins**. Each bin is assigned a label from **gnames**. The resulting categorization is stored back into the **y** variable.



Neural network model using the Keras

Here's a simplified breakdown of the steps:

1. Model Architecture: The model is built up of a number of thick layers that are piled on top of one another.
   * Every dense layer has a particular number of neurons and performs a particular activation function.
   * There are 13 neurons in the input layer (input\_shape=(13,)).
   * The ReLU activation function is used by 128, 32, 64, 64, and 64 neurons, respectively, in additional hidden layers.
   * One neuron in the output layer has a sigmoid activation function that makes it ideal for binary classification tasks.
2. Compilation: A learning rate of 0.0003 is used to build the model using the Adam optimizer.
   * The 'binary\_crossentropy' loss function, a standard choice for binary classification issues, has been selected as the loss function.
3. The model's effectiveness is evaluated using the accuracy metric.Training: The fit() method is used to train the model.
   * Target labels that correlate to the training data (scaledtrain) are provided.
   * The model has 100 training epochs.
   * Scaledtest and ytest data from validation are used to assess the model's performance during training.
   * The hist variable contains the training history, including loss and accuracy scores for each epoch.The code defines and trains a neural network model with multiple dense layers for binary classification. The model is compiled with an optimizer, loss function, and metrics, and then trained on the provided training data.

On the training set, we have 99% accuracy, but only 90% accuracy on the test set. The model's accuracy is comparatively excellent.

* 1. **Conclusion**

The Keras Neural Network accuracy is relatively high while the other machine learning model scores at 0.65 for Random Forest, 0.67 for Gradient Boosting Machine and 0.58 for K-Nearest Neighbors. Encoding the label for Keras Neural Network gives the advantage to the neural network in comparison while the other machine learning models has 5 labels to work with. It is a pretty successful exercise while developing the models as the models shows great improvement from their initial predictions.