Logo

Description automatically generated

**Practical assignment 2**

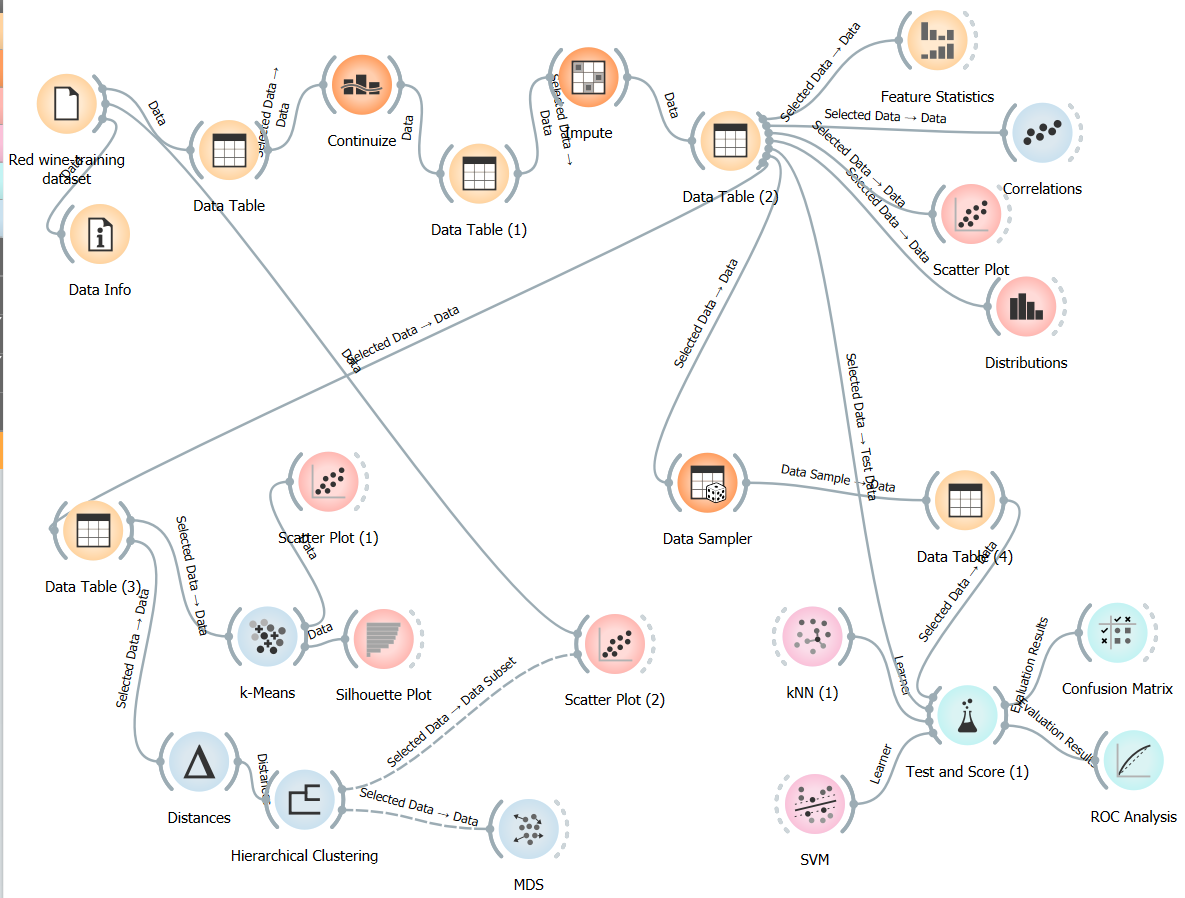
Practical Assignment 2

Student: Simeon Flamuraj

Student ID: 230ADB027

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

* **Orange Tool workflow:**



## Part I

* Dataset:

Title: Wine Quality

Source: Created by: Paulo Cortez (Univ. Minho), Antonio Cerdeira, Fernando Almeida, Telmo Matos and Jose Reis (CVRVV) @ 2009

Data Set information:

The two datasets are related to red and white variants of the Portuguese "Vinho Verde" wine. Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available (e.g. there is no data about grape types, wine brand, wine selling price, etc.)  
These datasets can be viewed as classification or regression tasks. The classes are ordered and not. Outlier detection algorithms could be used to detect the few excellent or poor wines.

Past usage:

In the above reference, two datasets were created, using red and white wine samples.

The inputs include objective tests (e.g. PH values) and the output is based on sensory data

(median of at least 3 evaluations made by wine experts). Each expert graded the wine quality

between 0 (very bad) and 10 (very excellent). Several data mining methods were applied to model

these datasets under a regression approach. The support vector machine model achieved the

best results. Several metrics were computed: MAD, confusion matrix for a fixed error tolerance(T),etc. Also, we plot the relative importance of the input variables (as measured by a sensitivity analysis procedure).

Attribute information:

The type of the data is a number,

Input variables (based on physicochemical tests):

1 - fixed acidity

2 - volatile acidity

3 - citric acid

4 - residual sugar

5 - chlorides

6 - free sulfur dioxide

7 - total sulfur dioxide

8 - density

9 - pH

10 - sulphates

11 - alcohol

Output variable (based on sensory data):

12 - quality (score between 0 and 10)

Missing attributes values: None

Number of instances: 4898

Number of attributes: 12

I have chosen to go with the red wine dataset.

Citation:

P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis.  
Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553, 2009.

Source:

Paulo Cortez, University of Minho, Guimarães, Portugal, <http://www3.dsi.uminho.pt/pcortez>  
A. Cerdeira, F. Almeida, T. Matos and J. Reis, Viticulture Commission of the Vinho Verde Region(CVRVV), Porto, Portugal  
@2009

**Visualization**:

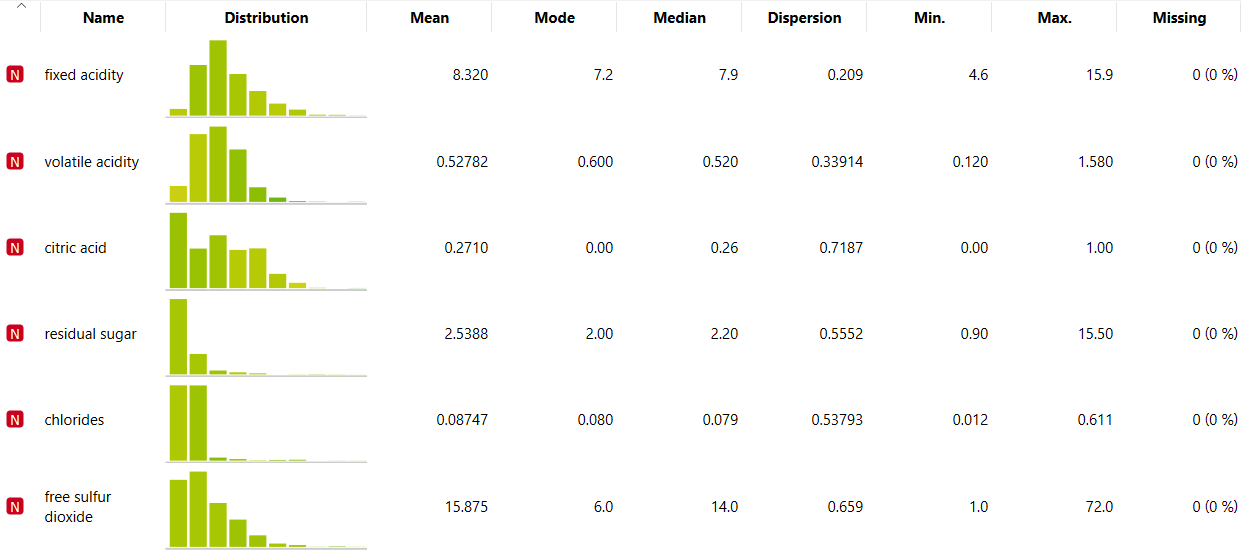
Feature statistic of the features in the dataset are as follows:  


Figure 1- 1st part of feature statistics widget

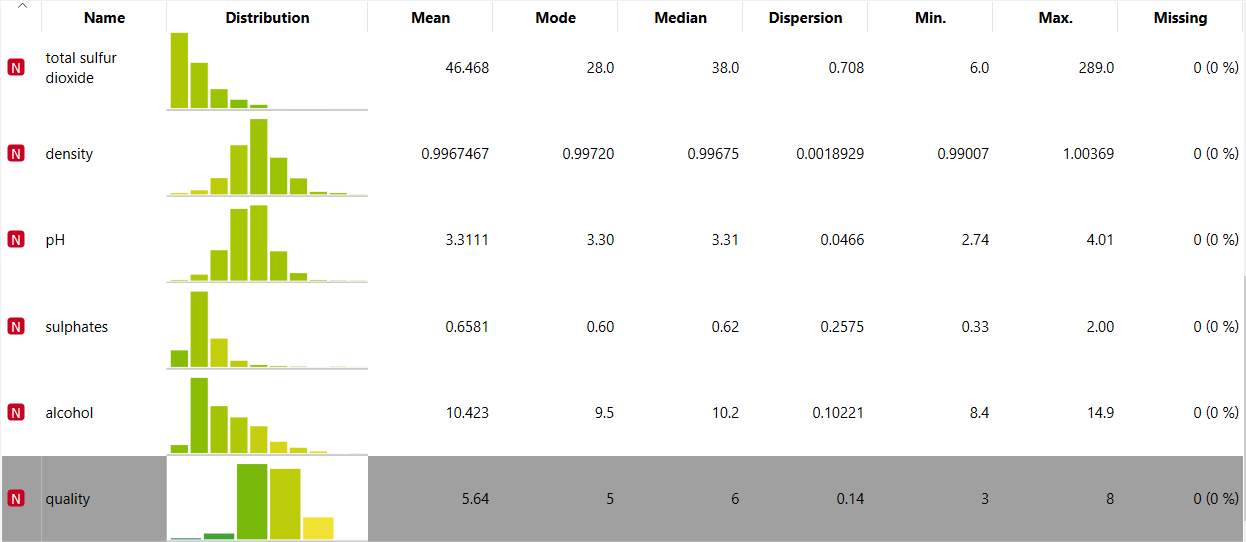


Figure 2-2nd part of feature statistics widget where quality is grey due to being the target

From the widget in orange tool named “Scatter Plot” I created two scatter plot which had more informative projections through an option in the widget and the color is about the quality as therefore is the output and target:

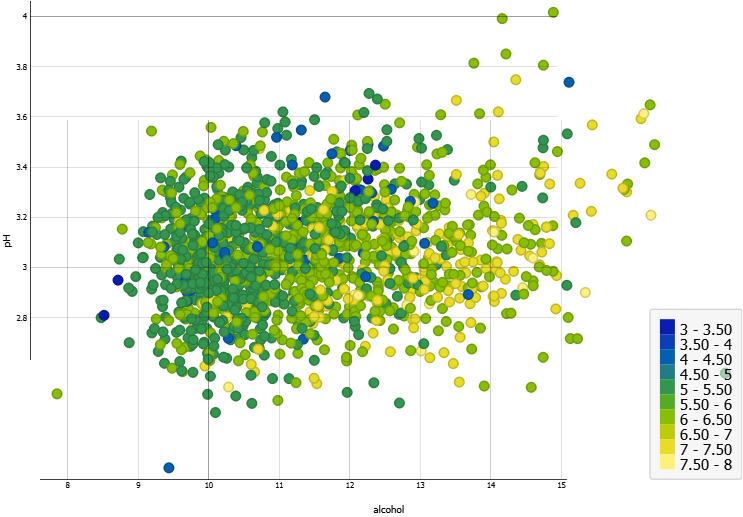
The first Scatter Plot is between alcohol and pH:

Figure 3- Scatter Plot between alcohol and pH

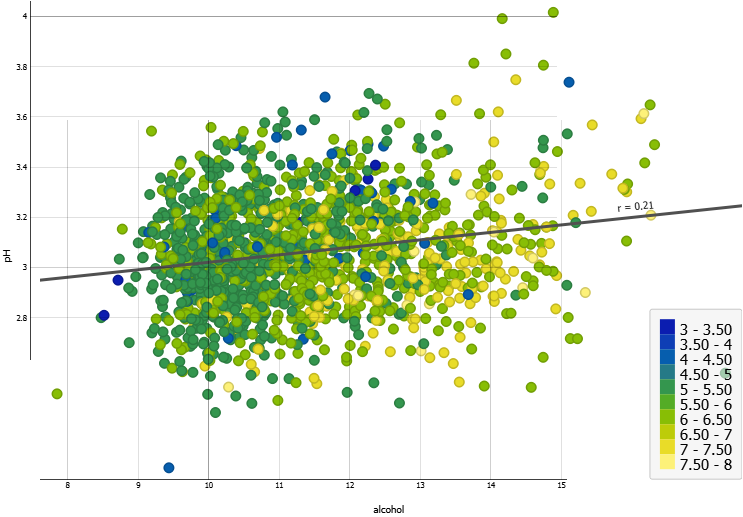
For this Scatter Plot the regression is r=0.21 and the regression line looks like below:

Figure 4- Scatter Plot between alcohol and pH with regression line

This means that there is a positive correlation between those two variables. The slope of the regression line quantifies the rate of change in the pH for a unit change in the alcohol. A slope of 0.21 suggest that for every unit increase in alcohol, the pH is expected to increase by 0.21 units.

2. The other scatter plot is between the pH and fixed acidity:

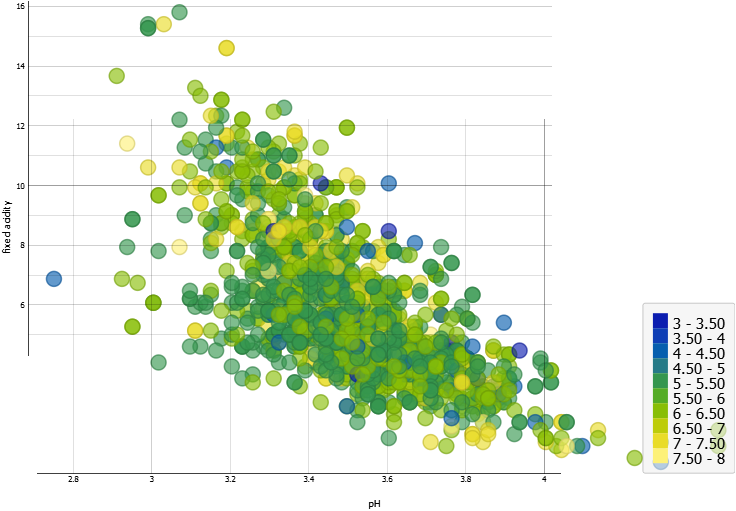


Figure 5- scatter plot between the pH and fixed acidity

For this scatter plot the regression is r= -0.68

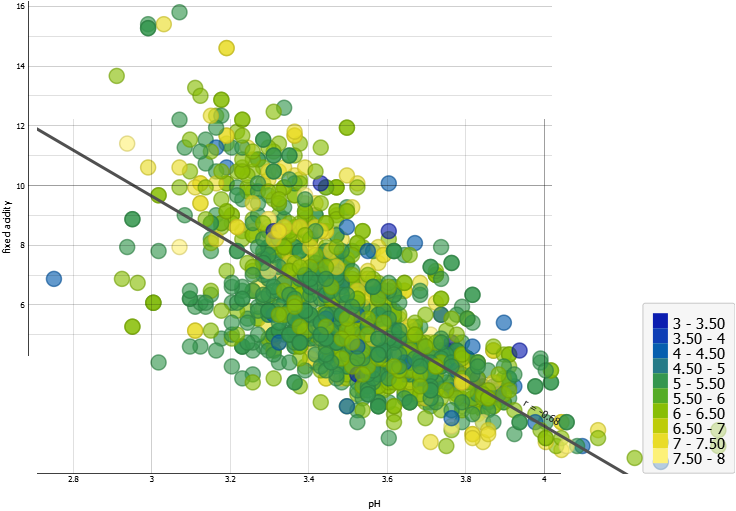


Figure 6- scatter plot between the pH and fixed acidity with regression line

This indicates a negative correlation between these two variables. The slope of -0.68 suggests that as pH increase the fixed acidity tends to decrease and vice versa. This implies that the wines with higher pH values generally have lower fixed levels.

This are two different scatter plots, in this case the dataset has 11 features and 1 target so we could create a lot more scatter plots. The colors in the scatter plots shows the quality of the wine which is the target and the output in attributes of dataset.

**Histograms:**

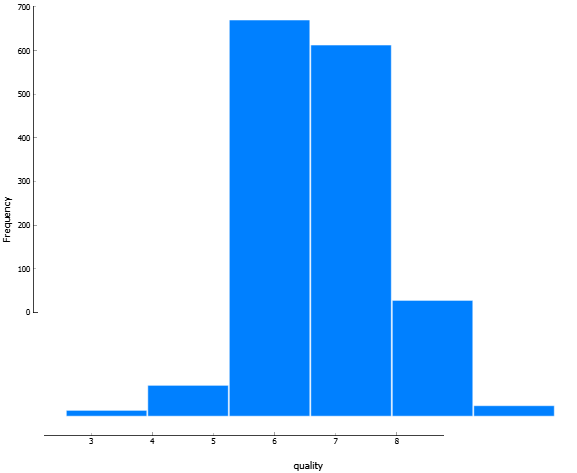


Figure 7- the frequency of the wine quality

In this histogram shows the frequency of the wine quality where 6 and 7 has the more frequency than the other qualities while the frequency of the quality less than 3 and more than 8 is 0 which means that the red wine quality is good and not bad for the most part of the tested samples of the red wine.

The histogram below shoes the frequency of alcohol in red wine where the most samples have a percentage of alcohol between 9 and 11. This shows that the wine alcohol from the samples gathered will be a good moderate quality if the percentage is inside the interval 9 and 11 according also to the quality histogram where it shows that most of the samples are a good quality where in scale of 10 they are 6 and 7 and a few 8.

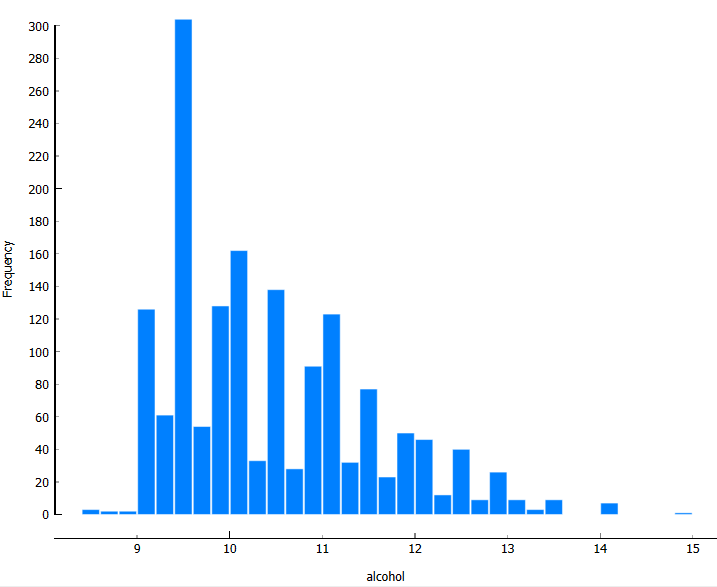


Figure 8- frequency of alcohol in red wine

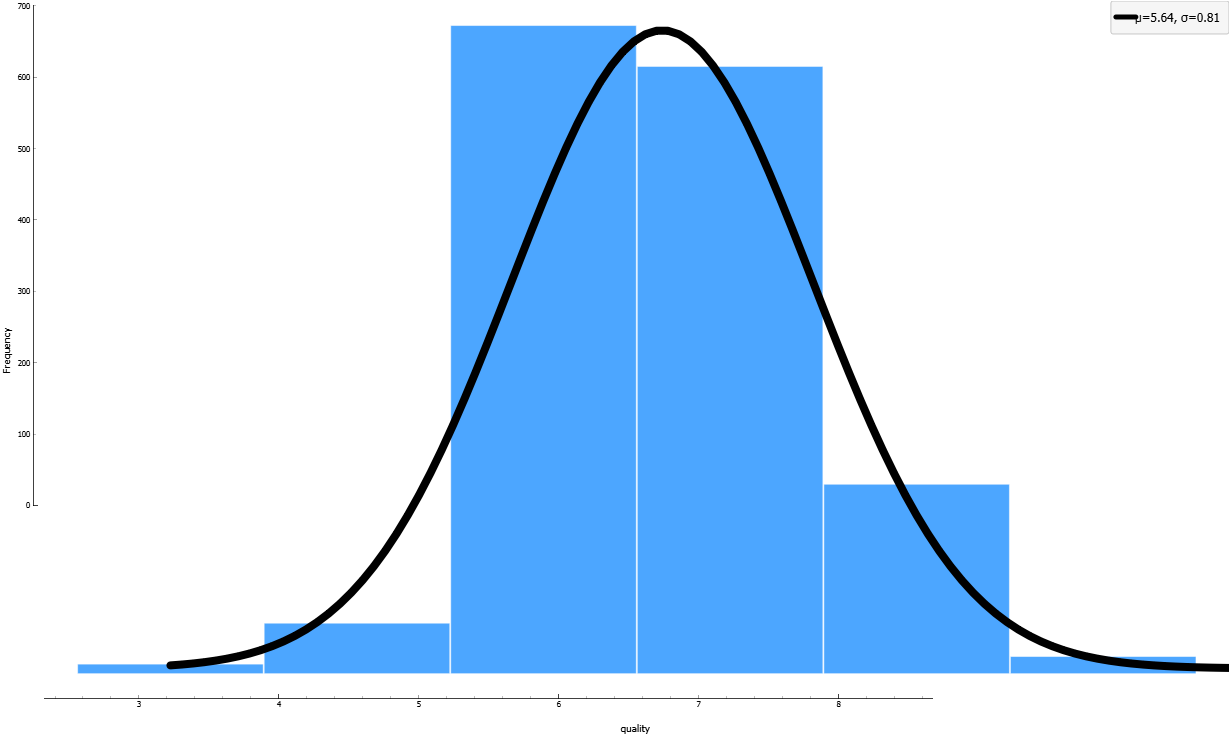
**Distribution:**  
this distribution below is for the quality of the red wines in dataset:  


Figure 9- distribution of the quality

So the distribution of the quality of has these characteristics µ=10.423 and σ=1.06

This is the distribution of alcohol

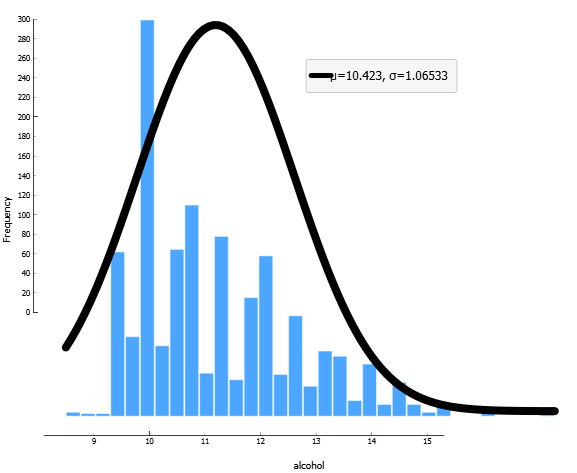


Figure 10- distribution of alcohol

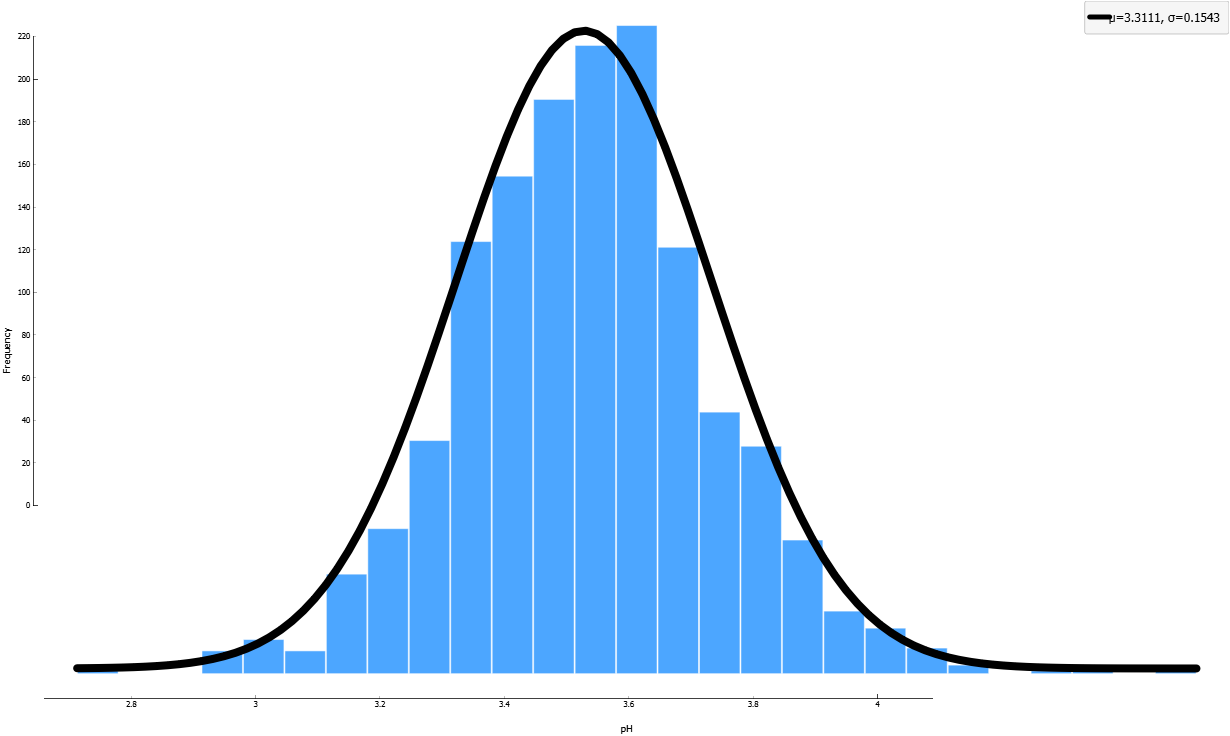
And the pH distribution :  


Figure 11-distribution of pH

**Conclusion:**

In this dataset there are a lot of features which impacts the quality of the red wine but I chose to visualize some key factors which impacts the quality like pH value, alcohol percentage. The scatter plot helps to find the regression line which then I found the slope point or the value of the regression line where in the regression line between the pH and alcohol was positive which in generally tells that in increase of pH also there is an increase of the alcohol percentage and via versa, meanwhile the regression in the scatter plot between the fixed acidity and pH was decreasing which means that the slope is negative which tells us that generally an increase of pH(x-axis) there is waited to be an decrease of fixed acidity(y-axis) and via versa. The distribution tells us that the distribution of the quality is almost in the middle where the quality is in the scale 1 to 10 and in the middle there are qualities 6, 7 and 8 but the most distribution is in 6 and 7 where there is waited to be no distribution in the range of qualities less than 3 and more tha 8. When we see the distribution of alcohol the most distribution is between 9 and 11 percentage but the difference between the quality distribution is the fact that there are more distribution in other values of alcohol. Form that I saw the distribution of the pH where the most distribution is in the middle of the histogram which ranges from 3 to 3.6. From those data we could say that the probability for the red wine is that the quality will be good where the alcohol is going to be around 9 to 11 percentage and the pH will be in range of 3 to 3.6.

Furthermore the dataset shows that the quality tends to distribute in the middle including pH, density, fixed acidity and volatile acidity which shows that acidity and pH impacts more the quality of the wine. Meanwhile citric acidity, residual sugars, chlorides, free sulfur dioxide, total sulfur dioxide, sulphates and alcohol tend to distribute in the first part of their values. But in the scatter plot we also found out that alcohol is impacted by the value of pH where in the increase of pH is the increase of alcohol.

To conclude, studying the distribution of the features of the dataset the wine will probably be 6 or 7 out of 10. For the unsupervised and supervised learning I will use those features: fixed acidity, citric acid, density, pH and the target: quality as therefore those impacted the most in the quality of the red wine.

## Part II

* **Unsupervised learning**

For the unsupervised learning I used two different algorithms: 1.K-means and 2. Hierarchical Clustering. The workflow in the orange tool looks like this:

**For the first experiment with the K-means I used this hyperparameters:**

**Number of clusters:** 2  
**Optimization:** initialize with KMeans++, 10 re-runs limited to 200 steps

**Data**

**Data instances:** 1599  
**Features:** fixed acidity, citric acid, density, pH  
**Target:** quality

**Silhouette scores for different numbers of clusters**

|  |  |
| --- | --- |
| **2** | 0.368 |
| **3** | 0.278 |
| **4** | 0.247 |
| **5** | 0.234 |
| **6** | 0.239 |
| **7** | 0.242 |
| **8** | 0.243 |
| **9** | 0.247 |
| **10** | 0.237 |
| **11** | 0.234 |

From the scatter plot we got this:

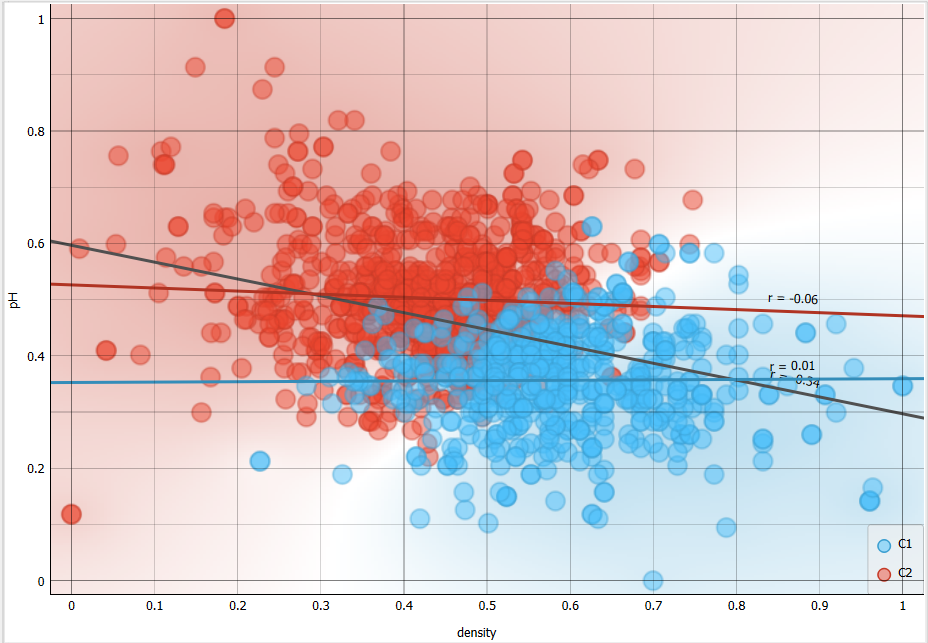


Figure 12-scatter plot of density and pH

From the scatter plot we could tell that the cluster are obviously separated from each-other expect in the middle where they form a group. From that there are created two lines of regression from the cluster shape with the slope value of -0.06, 0.01 and the main regression line acter the K-mean in the scatter plot of pH in y-axis and density in x-axis is -0.34. Within each cluster that the K-means algorithm has discovered, the regression lines demonstrate the correlations between pH and density. Insights into the patterns and structure contained in the data are provided by the various slopes, which imply changes in the intensity and direction of the association among the clusters.

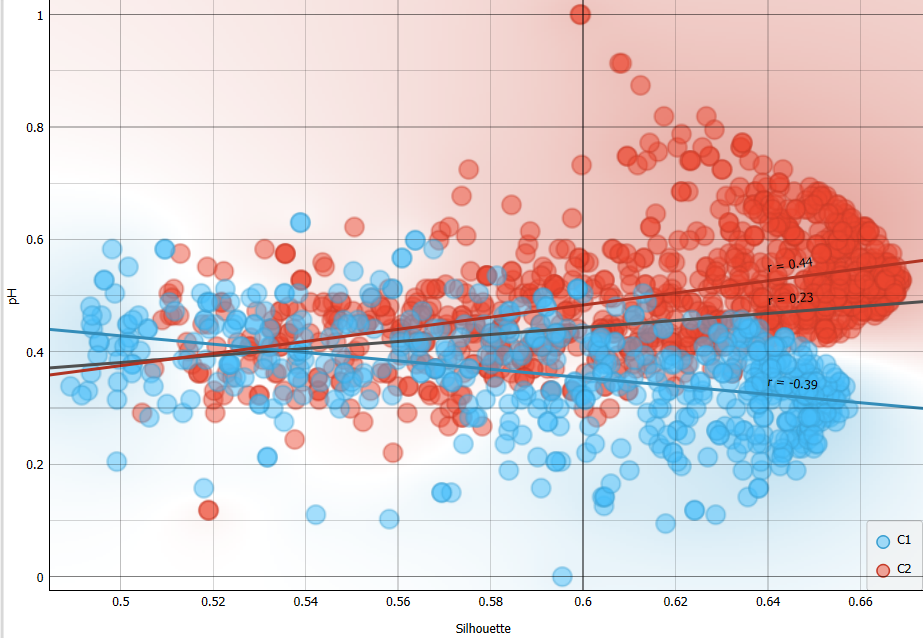


Figure 13-scatter plot of silhouette and pH

But in the scatter plot between the pH in y-axis and the silhouette in x-axis shows that the clusters are a little bit more mixed with the slope values as follows: for the blue line which is the first cluster the value of slope is -0.39 and for the second cluster with the color red the slope value is 0.44 and the value of the slope value for the regression line of pH and silhouette after K-mean is 0.23.

**For the second experiment I used those hyperparameters:**

**Number of clusters:** 2  
**Optimization:** initialize with KMeans++, 500 re-runs limited to 5000 steps

**Data**

**Data instances:** 1599  
**Features:** fixed acidity, citric acid, density, pH  
**Target:** quality

**Silhouette scores for different numbers of clusters**

|  |  |
| --- | --- |
| **2** | 0.368 |
| **3** | 0.278 |
| **4** | 0.247 |
| **5** | 0.233 |
| **6** | 0.240 |
| **7** | 0.239 |
| **8** | 0.239 |
| **9** | 0.248 |
| **10** | 0.233 |
| **11** | 0.239 |

For this experiment we see small changes to the Silhouette scores , those changes doesn’t impact in the change of the regression lines and the value of the cluster slopes or to the distributions. But we can argue that the K-Mean with 500 re-runs is more accurate.

**For the third experiment I used those hyperparameters:**

**Number of clusters:** 5  
**Optimization:** initialize with KMeans++, 500 re-runs limited to 5000 steps

**Data**

**Data instances:** 1599  
**Features:** fixed acidity, citric acid, density, pH  
**Target:** quality

Now we do not have Silhouette scores because we set a fixed number of clusters which are 5 clusters,

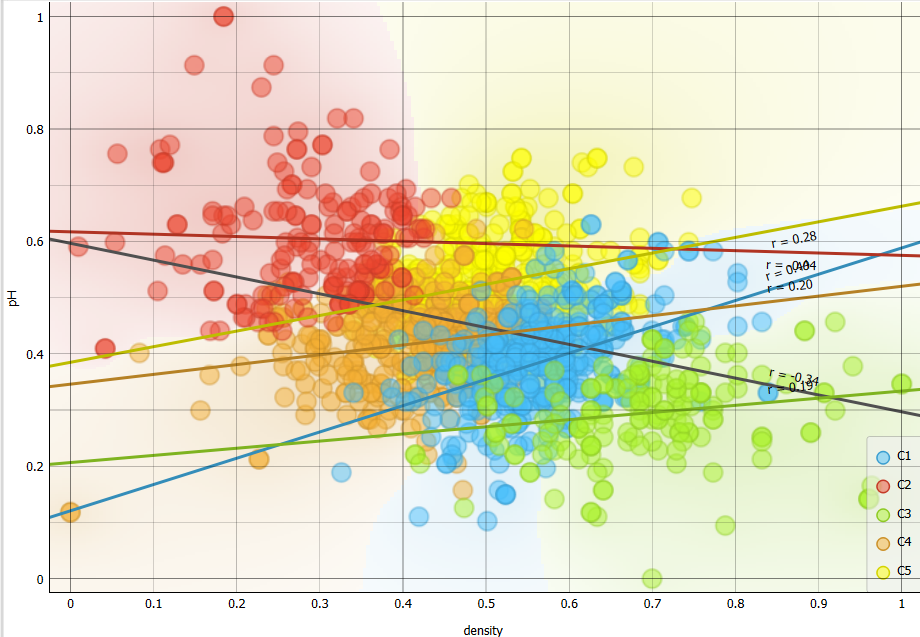
The scatter plot of pH and density now looks like this:  


Figure 14-new scatter plot of pH abd density

Now the slope value of the cluster 1 is 0.44, for the cluster 2 is -.0.04, for the cluster 3 is 0.19, for the cluster 4 is 0.20 and for the cluster 5 is 0.28. For the pH and density the slope is the same -.0.34

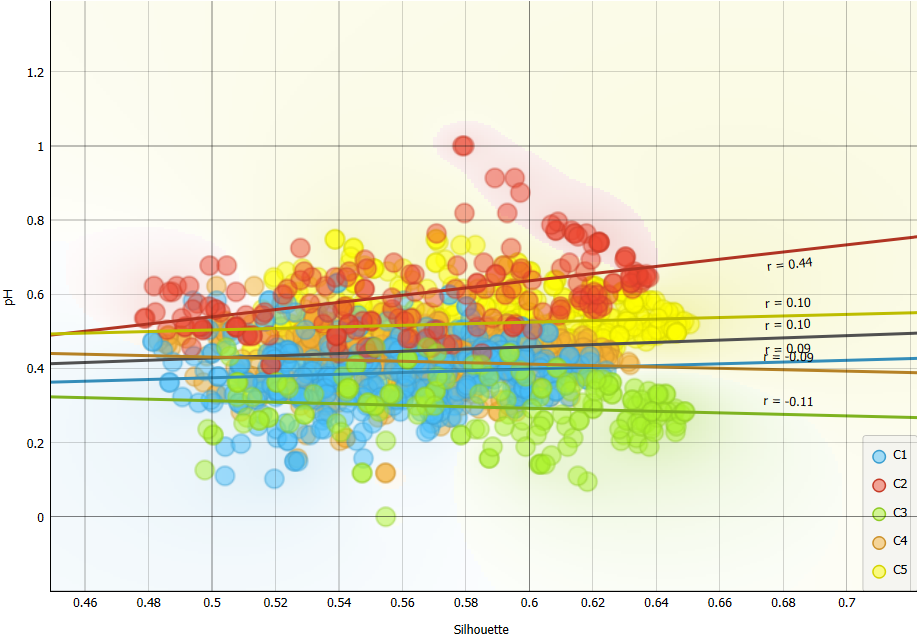
But the scatter plot of pH and Silhouette scores, even though we don’t have scores but we have values, looks like this with 5 clusters:  


Figure 15- scatter plot of pH and Silhouette scores

We could see a change of the slope value of regression line of pH and Silhouette which now is 0.09 but at other experiments was0 .23

We see that the K-mean is more accurate when has more re-runs and not fixed clusters.

For testing the performance of K-mean I used the widget Silhouette plot which was as follows:

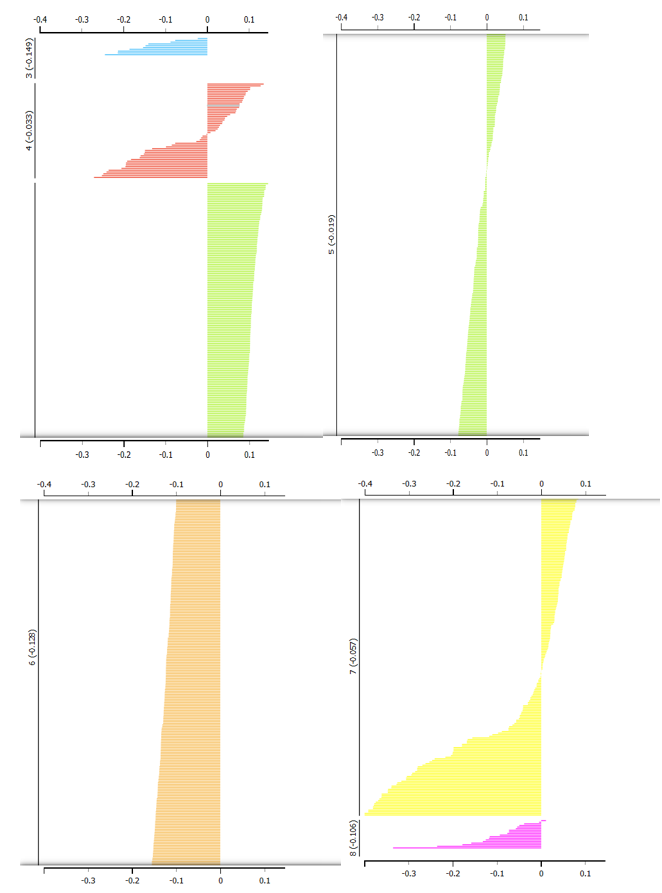


Figure 16- the widget Silhouette plot

For the algorithm hierarchical we also tried 3 experiments as follows:

1. For the first experiment I used distance compare by rows and Euclidian distance metric

And chose to have that group of data clustered together. For visualizing the data I used the MDS widget and with target quality and scatter plot connected to the hierarchical and original data:

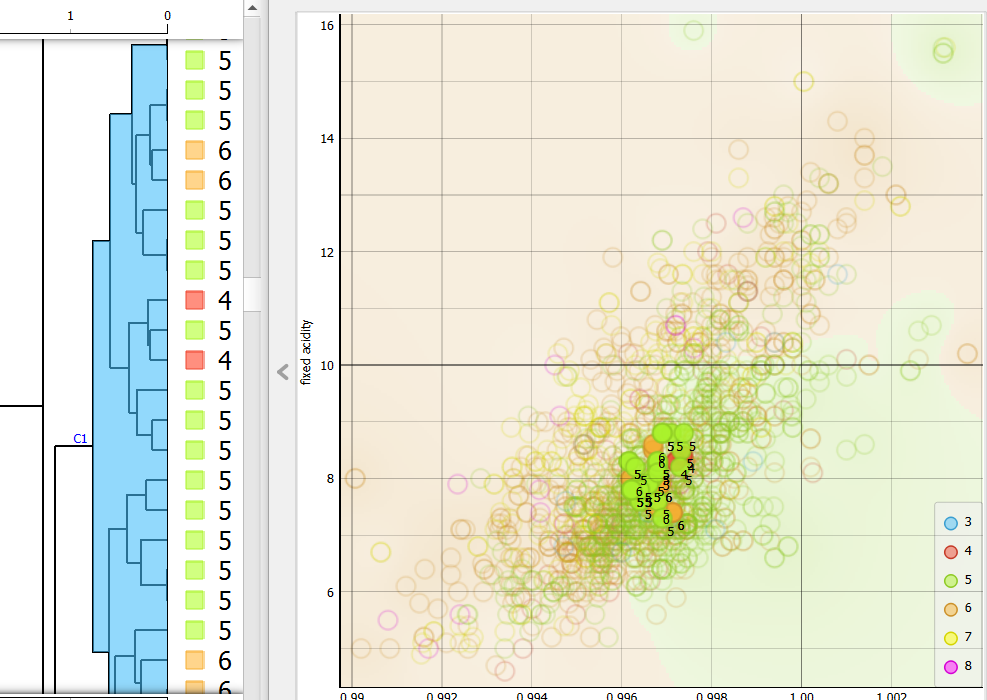


Figure 17-First group of cluster and scatter plot

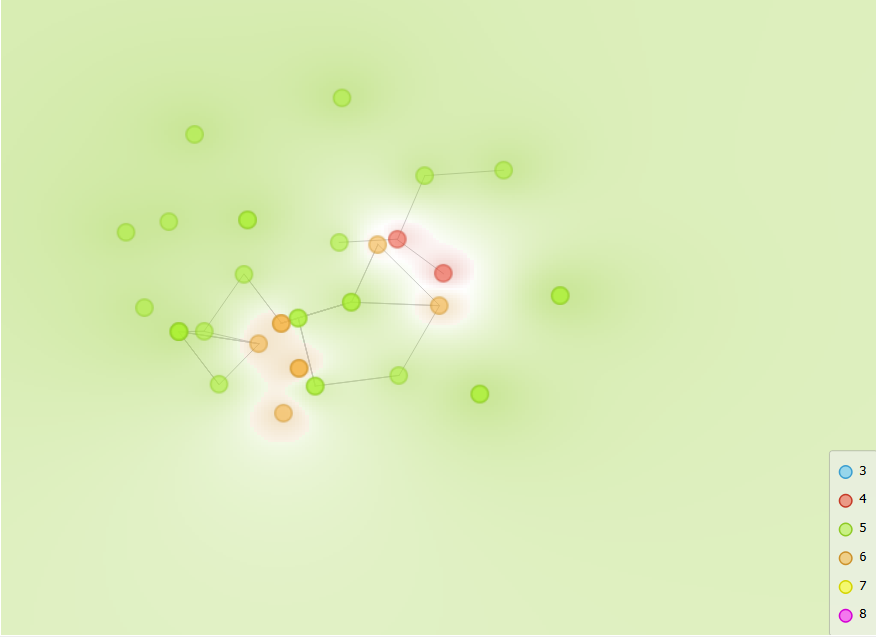


Figure 18-MDS widget

For the second experiment with this method I used another bigger cluster group in dendrogram and these were the results, the dendrogram is not full due to being not able to fit:

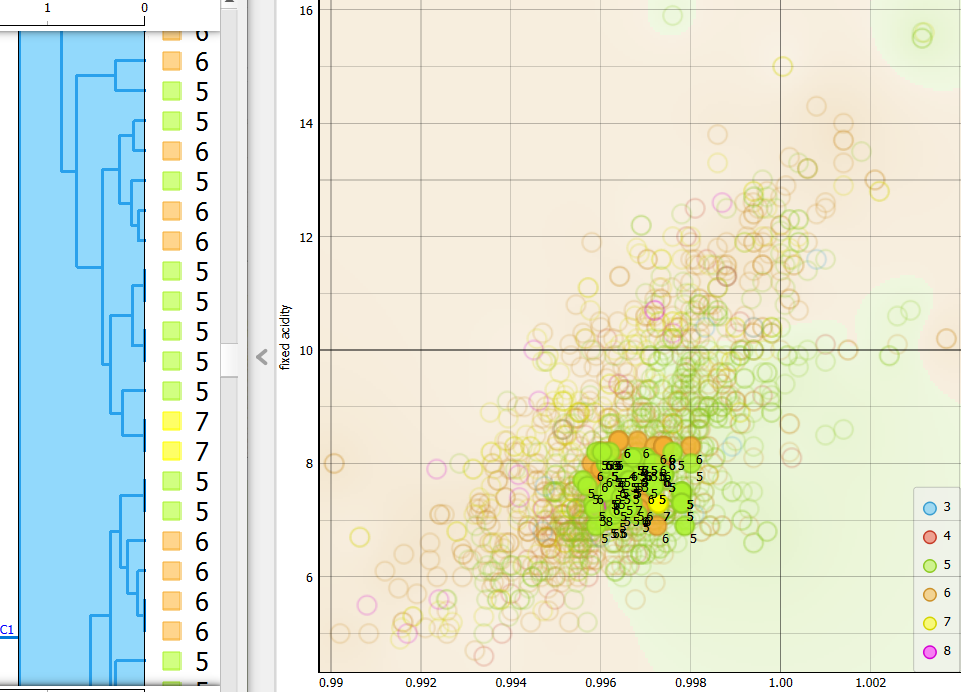


Figure 19--Second group of cluster and scatter plot



Figure 20-MDS widget

In this visualization we can see that in this cluster the group is bigger and also includes the previous experiment group in it.

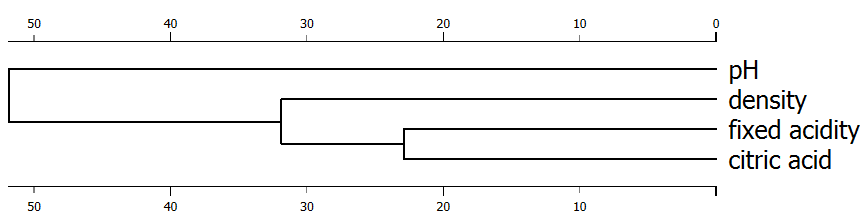
For the third experiment I thought to change things a bit, in distance I let the distance metric the same but I changed compare from row to column and this was the result:  


Figure 21-third group of clusters

In this case it has clustered the features that impacts the more In quality.

**Conclusion:**

To conclude for the unsupervised learning both methods the K-mean algorithm and hierarchical clustering can offer a way to find the quality of the red wine bot both have their strength and weaknesses.

K-Mean is impacted by hyperparameters. As more re-runs as more it gets improved, as shown the silhouette points have had an impact by changing the re-runs and the limitation of re-runs in the K-Mean options. Based on that score the methods created regression lines to see the connections of the features and than implied to predict the quality.

Hierarchical Clustering, uses the distance Metrix to see the relationships and forms group which in some ways it can predict faster. But in the groups is not always the same values, as shown that there are some different values grouped together, this might give a not near right prediction.

Comparing both methods, the best methods in my idea is K-Means because in right hyperparameters can predict the quality better than hierarchical clustering.

## Part III

* **Supervised Learning**

For the supervised learning I chose to use these two methods: kNN algorithm and SVM algorithm. Both are used for supervised learning and fulfill the objective to work on classification. To the why I chose those methods are that the first methods kNN was talked during class and had manual about it and for the SVM researched online.

First method kNN stands for k-Nearest Neighbors, is a supervised learning algorithm used for classification and regression tasks. It is a non-parametric algorithm, meaning it does not make any assumptions about the underlying data distribution. Instead, it makes predictions based on the similarity between a new data point and the labeled data points in the training dataset.

Second method SVM stands for Support Vector Machine, is a supervised learning algorithm used for classification and regression tasks. It is a powerful algorithm known for its effectiveness in handling high-dimensional data and its ability to find optimal decision boundaries.

So this two methods are good methods for supervised learning:

For the train data I used all the original data, but the setting widget for the sample data which generated data for the test are :

1.1


Figure 22-first hyperparameters for the method kNN

kNN methods:

The hyperparameters are: Wight-uniform, metric Euclidian and number of neighbors-5. And this are the result:  
For the train data:  


Figure 23-test and scores widget

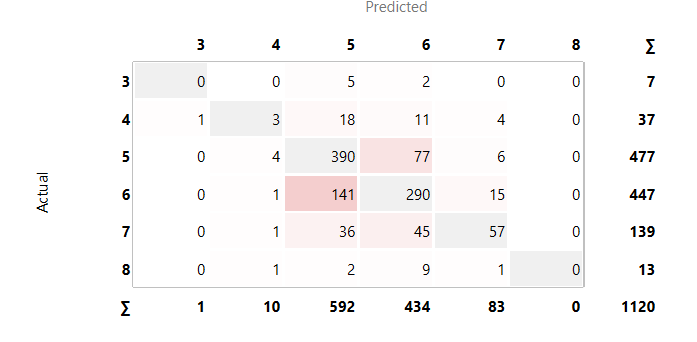


Figure 24-confusion metrix

For the test data:  


Figure 25-test and scores widget

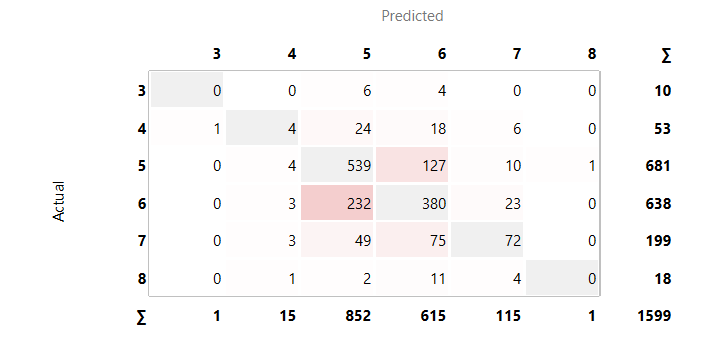


Figure 26-confusion metrix

For the second experiment I changed the wight from uniform to by distance and the changes were as follows:

For the train data:



Figure 27-test and scores widget

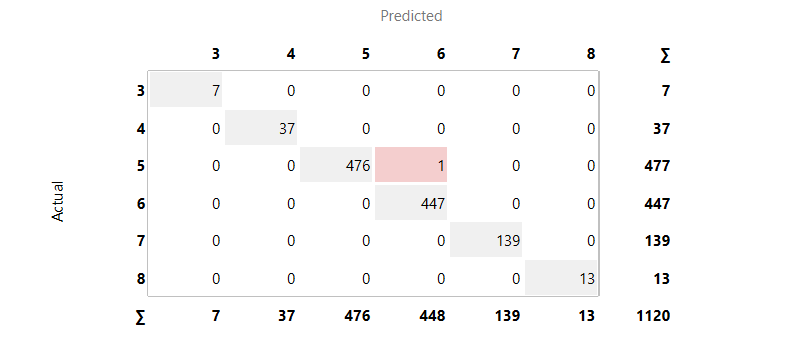


Figure 28- confusion metrix

For test data:  


Figure 29-test and scores widget

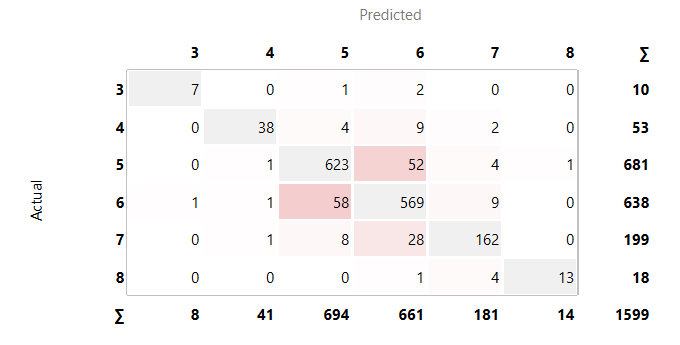


Figure 30-confusion metrix

And for the third experiment including the change in the second experiment I changed the neighbour value from 5 to 2 and the changes were as follows:

For the train data:  


Figure 31-test and scores widget

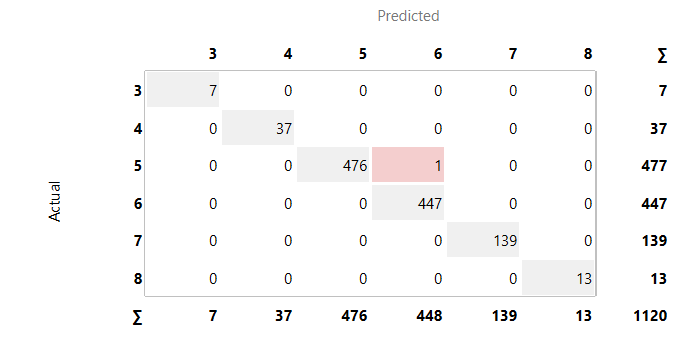


Figure 32-confusion metrix

For test data:



Figure 33-test and scores widget

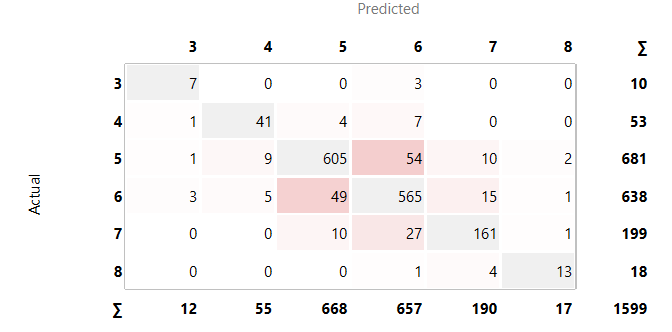


Figure 34-confusion metrix

Now the first experiment with the method SVM algorithm was done with these hyperparameters:

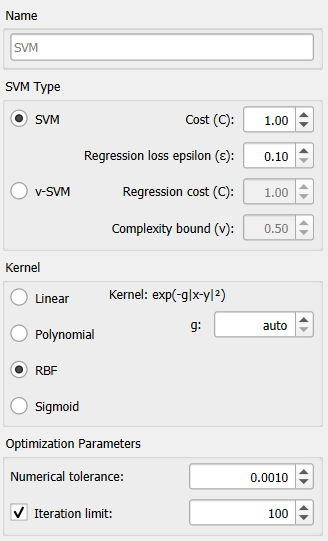


Figure 35-first hyperparameters of SVM

And the results were like below:  
For the train data:  


Figure 36-test and scores widget

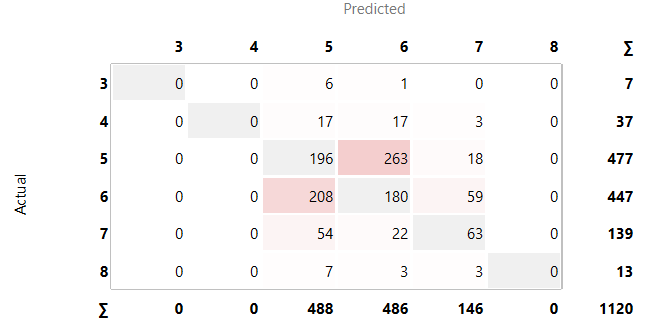


Figure 37-confusion metrix

For the test data:  


Figure 38-test and scores widget

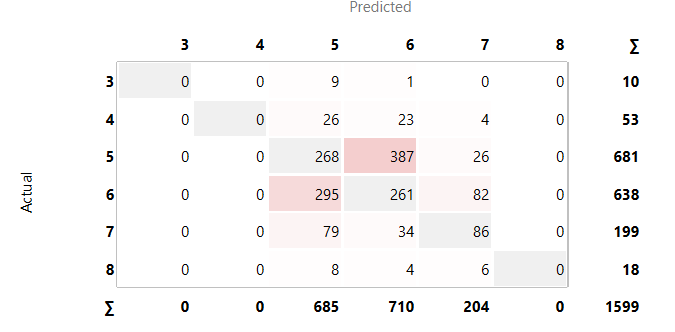


Figure 39-confusion metrix

For the next experiment in widget setting I changed the iteration limit from 100 to 200 and the changes were as follows:

For the train data:  


Figure 40-test and scores widget

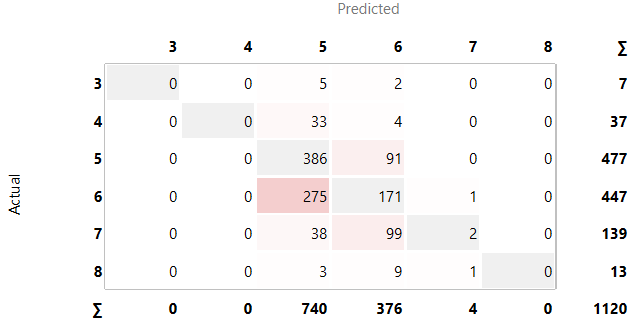


Figure 41-confusion metrix

For the test data:  


Figure 42-test and scores widget

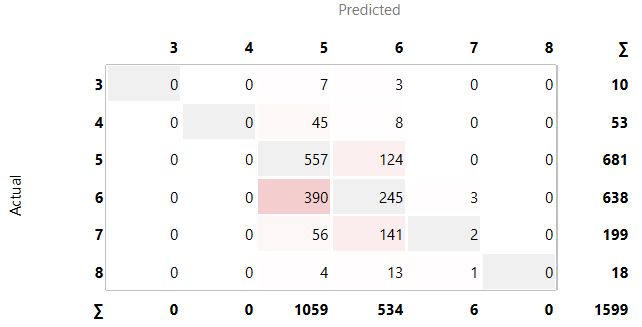


Figure 43-confusion metrix

And for the last experiment I kept the changes from the second experiment and I changed kernel from RBF to linear and changes were as follows:  
For train data:  


Figure 44-test and scores widget



Figure 45-confusion metrix

For the test data:  


Figure 46-test and scores widget

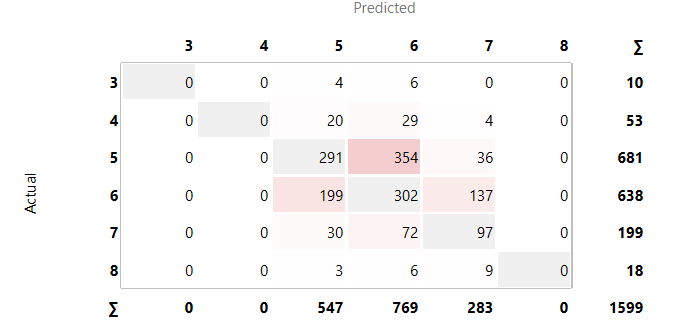


Figure 47-confusion metrix

To see a compare for the methods I compared the last experiment with KNN and SVM with ROC analysis:

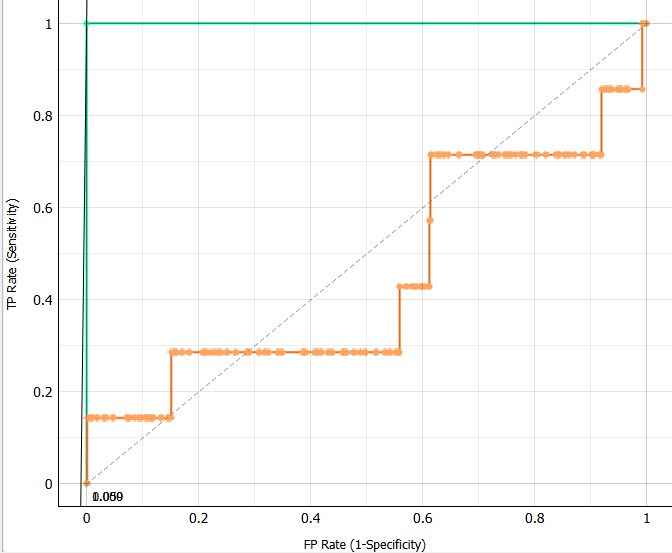
For the train data:  


Figure 48-ROC analysis widget

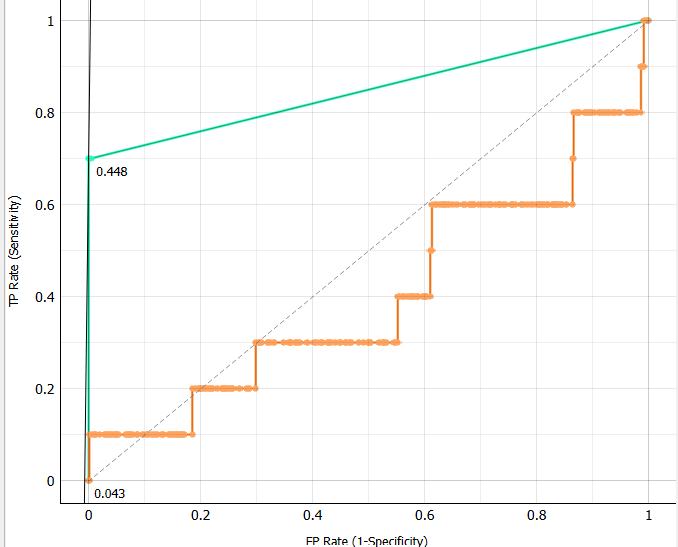
For the test data:  


Figure 49-ROC analysis widget