# **RED WINE QUALITY** **PROJECT**



**The article contains the following sub-topics**:

1.      Problem Definition  
2.      Data Analysis  
3.      EDA Concluding Remarks  
4.      Pre-processing Pipeline  
5.      Building Machine Learning Models  
6.     Concluding Remarks

# Problem Definition

The dataset is 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.).  
  
 This dataset can be viewed as classification task. The classes are ordered and not balanced (e.g. there are many more normal wines than excellent or poor ones). Also, we are not sure if all input variables are relevant. So it could be interesting to test feature selection methods.

**Attribute Information**

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)

What might be an interesting thing to do is to set an arbitrary cutoff for your dependent variable (wine quality) at e.g. 7 or higher getting classified as 'good/1' and the remainder as 'not good/0'.

**Objectives**

The objectives of this project are as follows

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

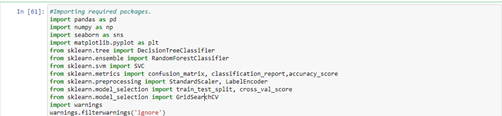
2. To determine which features are the most indicative of a good quality wine

3. This allows you to practice with hyper parameter tuning on e.g. decision tree algorithms looking at the ROC curve and the AUC value.

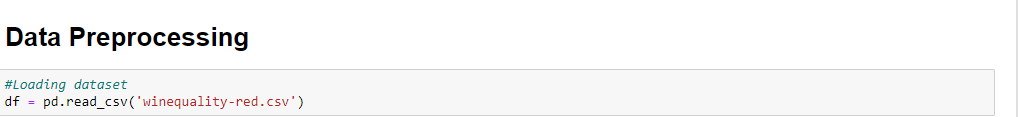
# Data Analysis

Exploratory data analysis is a very important step to get an idea of what the dataset is like and what kind of modifications we need to make. I started with importing the libraries and modules and reading the data I’ll be using into pandas dataframe

**Importing Libraries**

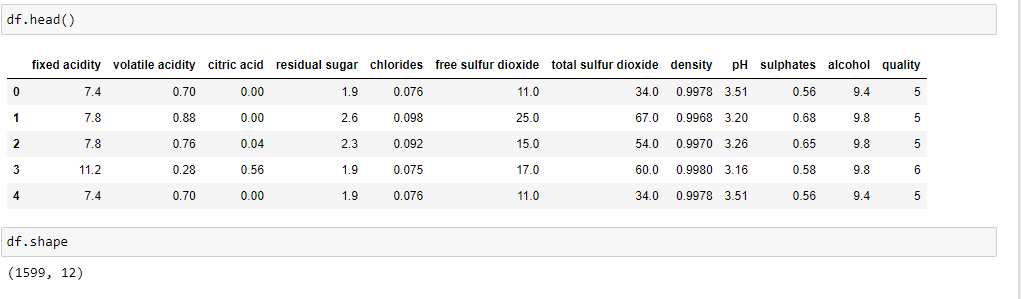


**Reading the Data**



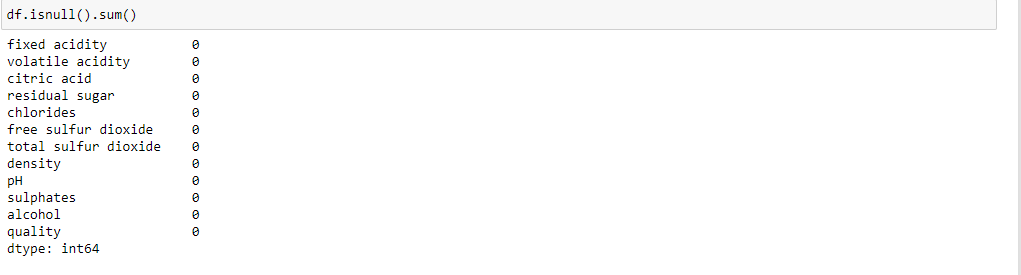
**Analyzing data**

**As we have loaded dataset, let’s see how many records are present. Also we can check number of rows and columns.**

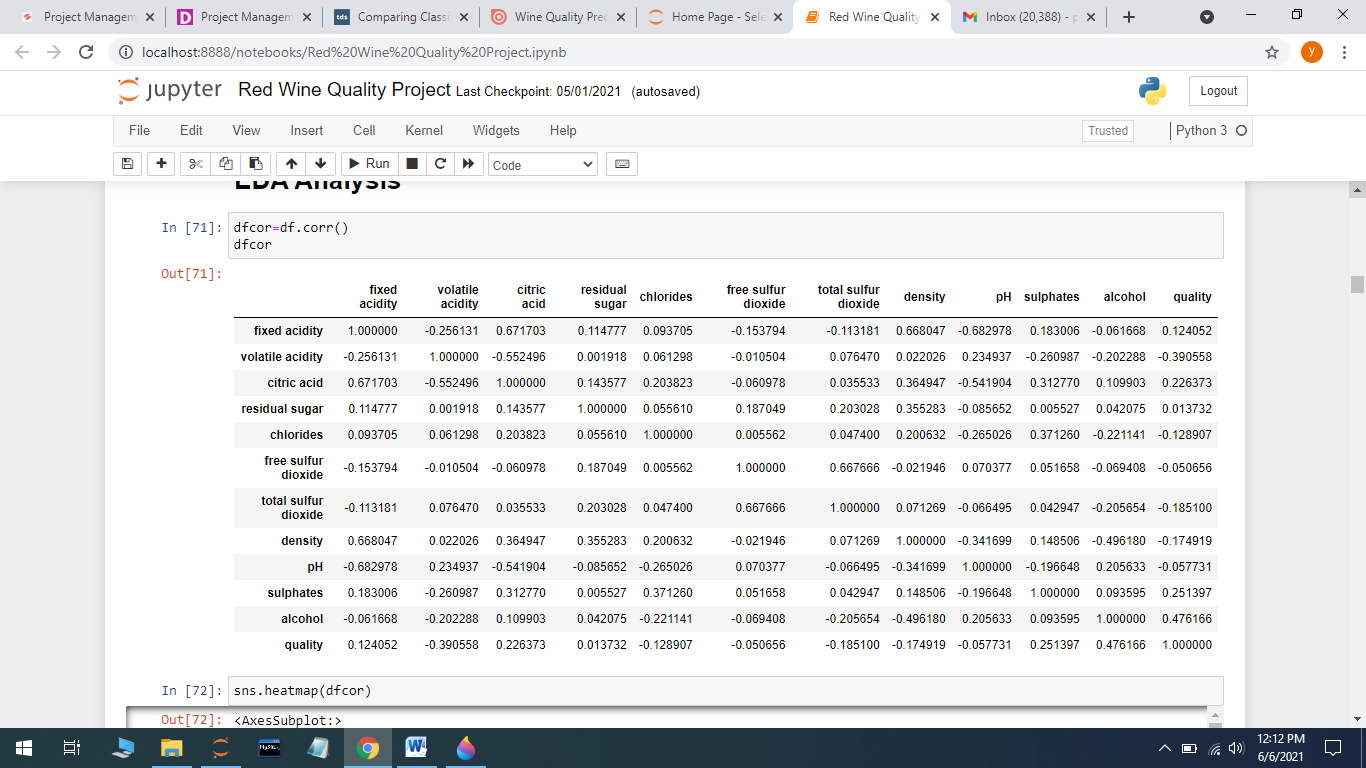
****

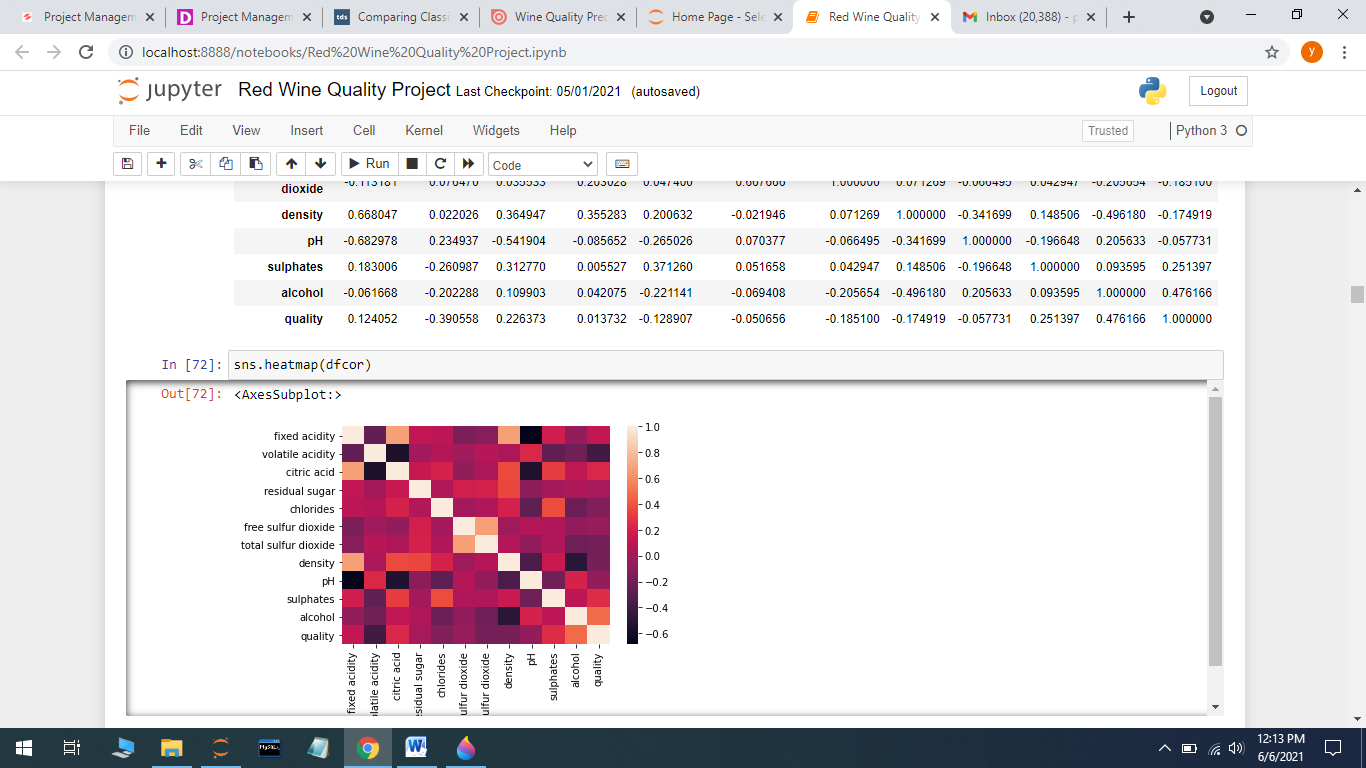
**Checking missing values**

There is no missing value in the data so it’s ready to analyze but first outliers must be detected and eliminated before the modeling step

****

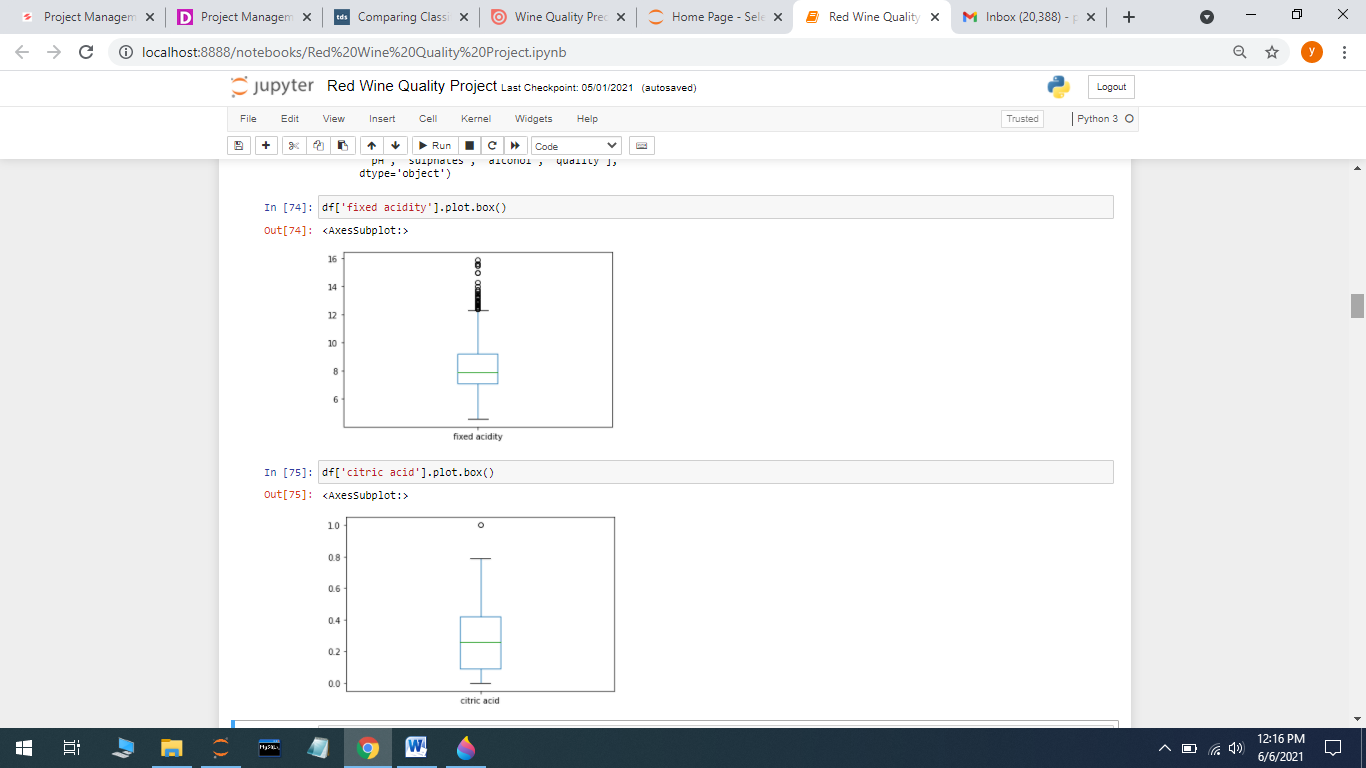
**Checking the correlation between columns**





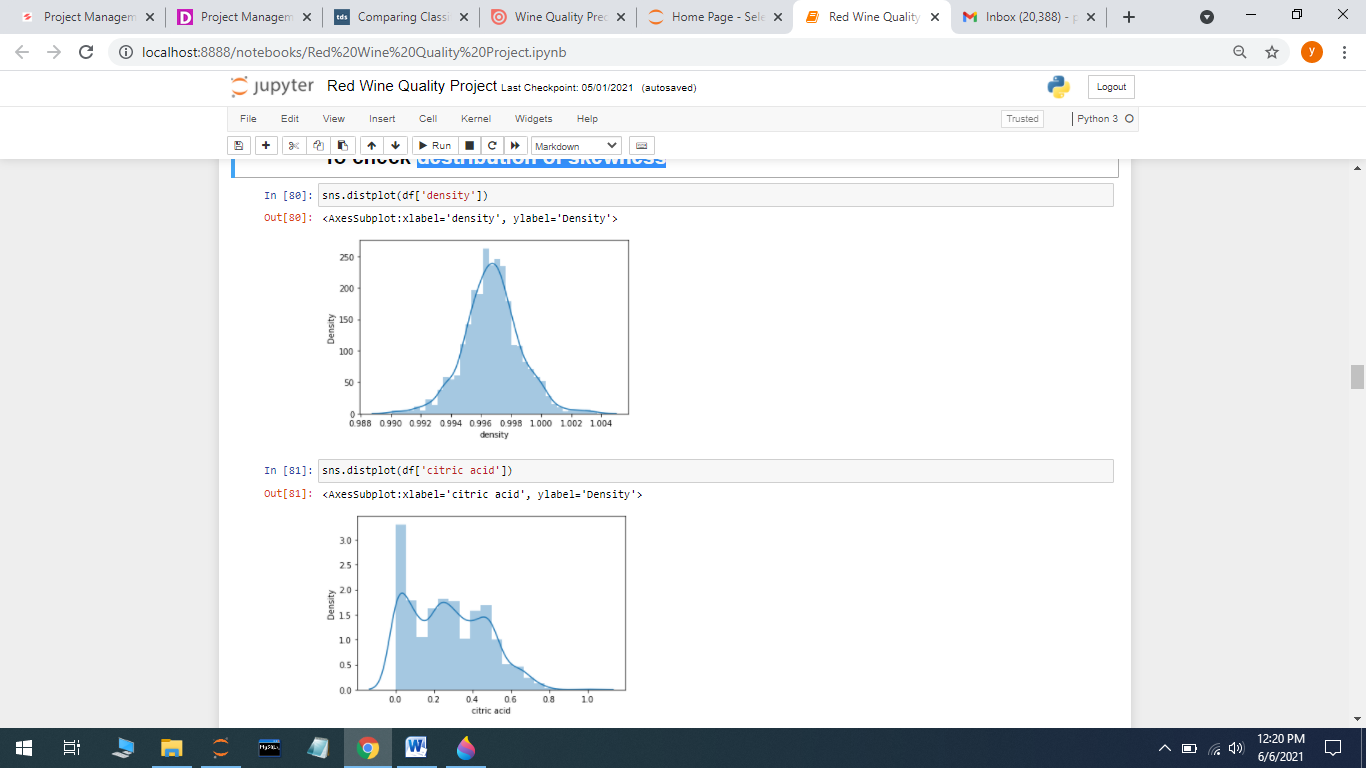
Dark color shades means those columns are highly correlated. white color shade means low correlated. Quality is highly correlated with alchohol. alchohol highly negatively correlated with density with 0.5.(see in dark maroon shade) density is highly positively correlated with residual sugar. volatile acidity is highly negatively correlated with quality. free sulfer dioxide is highly correlated with total sulfer dioxide. hence we can drop volatile acidity from dataset for further linear regression.When I checked the correlation between columns I can see that some of the features are strongly correlated with quality while some of them are not.

# Data Visualization



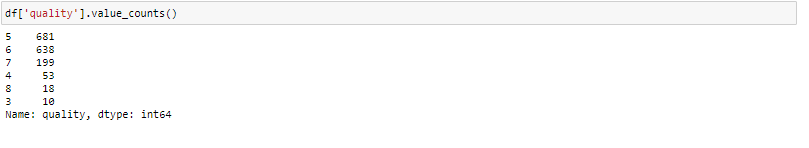
I used boxplot to visualize the distribution of the values in ‘fixed acidity’ and 'citric acid' columns to get a better insight. The actual maximum value is bigger than that are outliers since they are not included in the box of observation.

**Distribution of skewness**



As we can see ‘density’ column is showing normal distribution and ‘citric acid' is not showing normally distributed and its skewed.

**Checking the class imbalance**

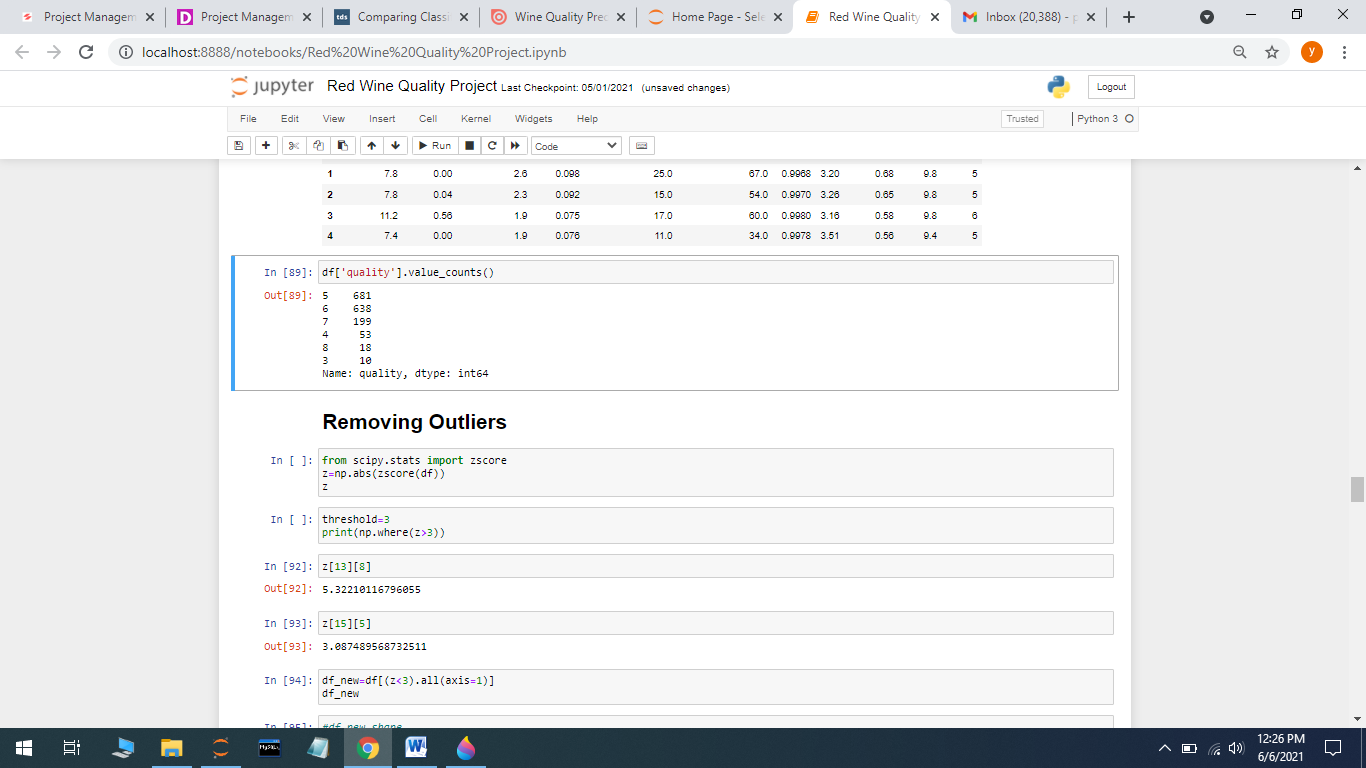


In this query we get count of values in ‘quality’ column in dataset. Hence we observe that many observations are belonging to max score i.e. 5,6,7.

**Eliminating outliers by using Z-score**

Most of the time it’s important to remove outliers since they would most likely affect the performance of machine learning models. But let’s say 30% of your dataset are outliers. Then it may not be wise to remove them all because probably there is something more going on and it needs to be inspected further. In order to find and remove the outliers, I used z-score.

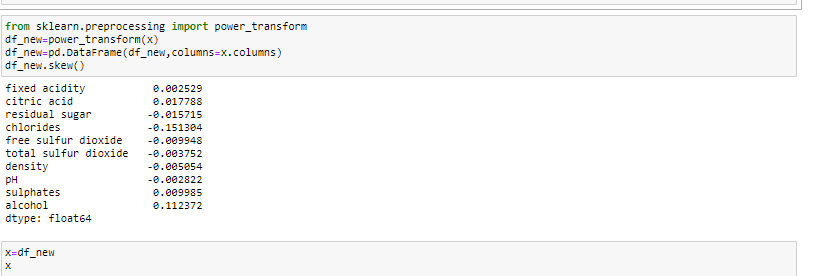
https://miro.medium.com/max/60/1*g1xhcuv3IHHm7EKUP5JkSA.png?q=20

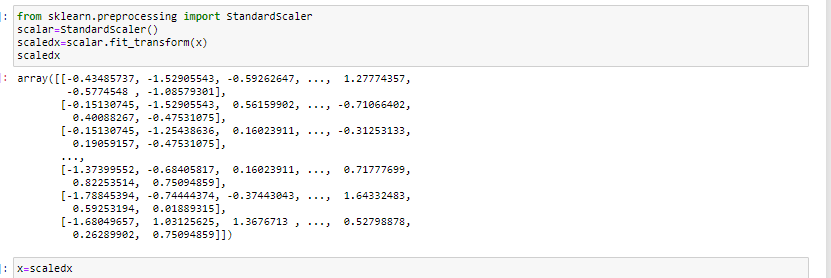


# Data Preprocessing

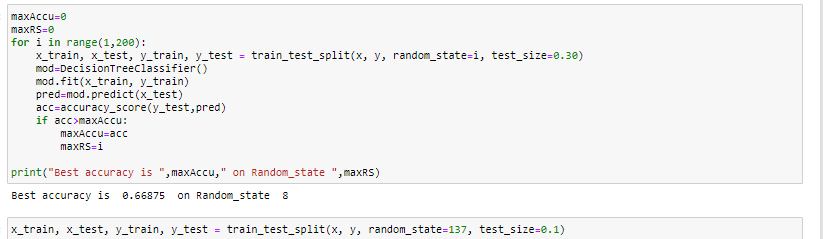
In this step of the analysis I defined the features to train and test the machine learning model and the target to predict. And then I did standardization(also called z-score normalization) for the features because different scales of features may impact the performance of the machine learning models.

For this purpose, I used StandardScaler() function defined in Scikit-learn. And finally I split the dataset into training and test sets 80% and 20% respectively.



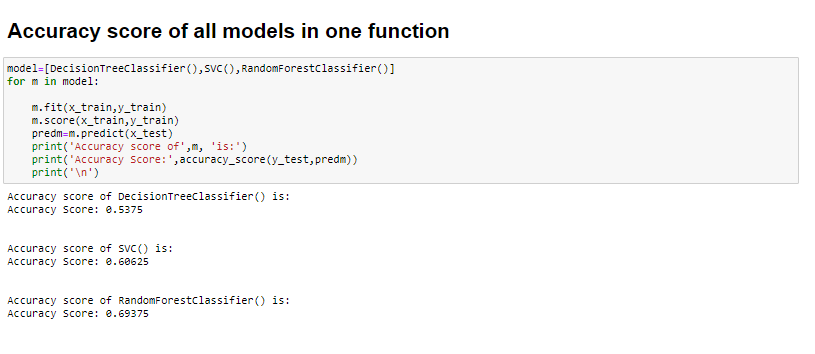


**Train and test sets split**

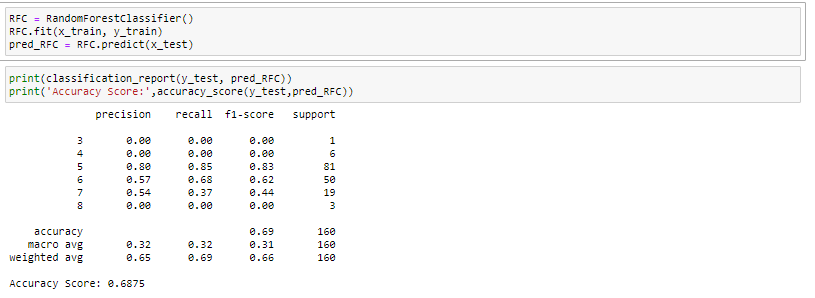
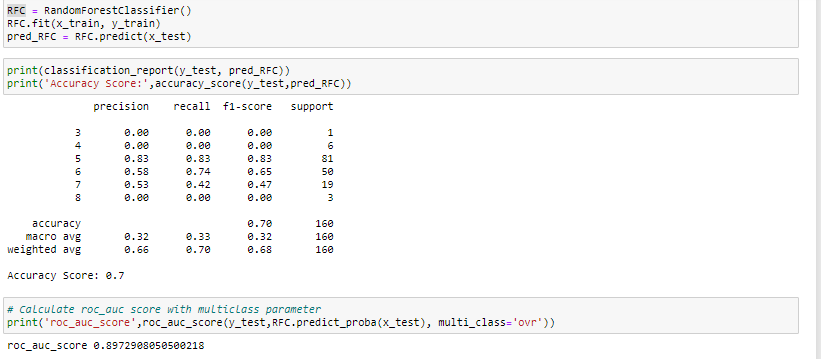


**Validation and Model Selection**

In this part I trained several classification algorithms to find the best one for the dataset I used.



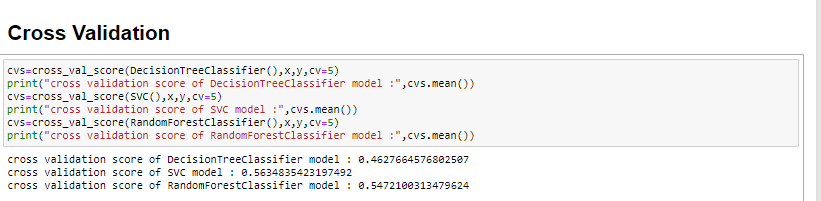
We can clearly see that RFC is giving the best accuracy score. Random forest is an ensemble learning method that builds multiple decision trees and then gets a prediction based on what the majority of decision trees predict.



**Cross Validation Score**

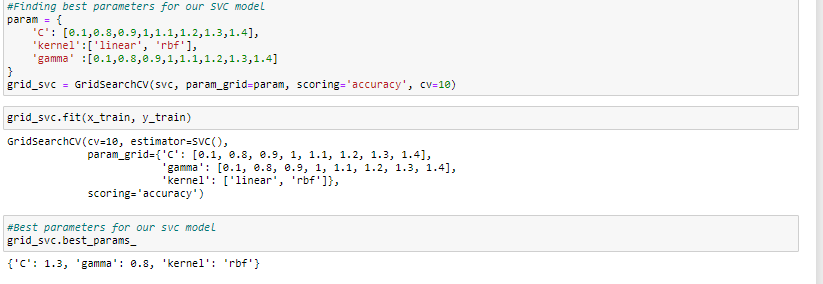
Cross-validation is a statistical method used to estimate the skill of machine learning models. ... That k-fold cross validation is a procedure used to estimate the skill of the model on new data.

After comparing accuracy score and cross validation score, SVC is the best model which gives more accuracy.

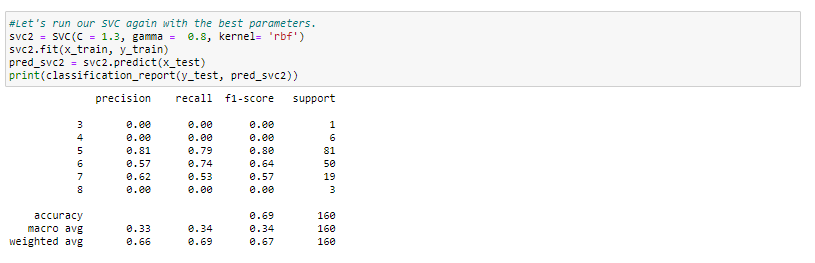


**Hyperparameter Tuning**

In machine learning, hyperparameter optimization or tuning is the problem of choosing a set of optimal hyperparameters for a learning algorithm. A hyperparameter is a parameter whose value is used to control the learning process. By contrast, the values of other parameters (typically node weights) are learned.



**Final Model**



# Conclusion

In this project, I used K-Nearest Neighbors, Logistic Regression with polynomial features, Decision Tree, and Random Forest. With the roc\_auc\_score in Scikit-learn I calculated the AUC score for each model. Also using cross\_val\_score method I found AUC score using cross validation method.

If we compare the cross validation scores and recall results of all models we can see that the best results were obtained with **SVC model.**