**WINE QUALTIY ANALYSIS**

**A Online Summer Internship project *report submitted to the VNR VIGNANA JYOTHI INSTITUE OF ENGINEERING AND TECHNOLOGY in partial fulfillment of the requirements for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**Submitted By**

**BHUKYA SHASHIDHAR (18071A0561)**

**CHINTA CHAKRAPANI (18071A0570)**

**KONGARA NIKHIL (18071A0584)**

**BOJJA KRANTHI (19075A0507)**

**UNDER THE GUIDANCE OF**

MRS. L. INDIRA

(Assistant Professor, VNR VJIET)



**COMPUTER SCIENCE AND ENGINEERING DEPARTMENT**

**VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY**

(An Autonomous Institute, NAAC Accredited With ‘A++’ Grade, NBA Accredited, Approved by AICTE, New Delhi, Affiliated to JNTUH)

SEPT,2020.

**VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY**

(An Autonomous Institute, NAAC Accredited With ‘A++’ Grade, NBA Accredited, Approved by AICTE, New Delhi, Affiliated to JNTUH)



**CERTIFICATE**

This is to certify that **B.Shashidhar (18071A0562), Ch.Chakrapani (18071A0570), K.Nikhil(18071A0584), B.Kranthi(19075A0507**)have successfully completed their Online Summer Internship projectwork at CSE Department of VNR VJIET, Hyderabad entitled “WINEQUALITY ANALYSIS ” in partial fulfilment of the requirements forthe award of B.Tech degree during the academic year 2020-2021.This work is carried out under my supervision and has not beensubmitted to any other University/Institute for award of anydegree/diploma.

Project Guide Head of the Department

**Mrs.L.Indira Dr.B.V.Kiranmayee**

Assistant Professor Associate Professor &

Department of CSE Head of the Department CSE

VNR VJIET VNR VJIET

DECLARATION

This is to certify that the project work entitled “**WINE QUALITY ANALYSIS”** submitted in VNR Vignana Jyothi Institute of Engineering & Technology in partial fulfilement of requirement for the award of Bachelor Technology in Computer Science and Engineering is Bonfide report of the work carried out by us under the guidance and supervision of Mrs. L. Indira (Assistant Professor), Department of CSE, VNRVJIET. To the best of our knowledge, this report has not been submitted in any form to any university or institution for the award of any degree or diploma

**B.Shashidhar Ch.Chakrapani K.Nikhil B.Kranthi**

(18071A0562) (18071A0570) (18071A0584) (19075A0507)

II B.Tech-CSE, II B.Tech-CSE, II B.Tech-CSE, II Btech-CSE,

VNR VJIET VNR VJIET VNR VJIET VNR VJIET

ACKNOWLEDGEMENT

Behind every achievement lies an unfathomable sea of gratitude to those who

activated it, without it would ever never have come into existence. To them we

lay the words of gratitude imprinting within us.

We are indebted to our venerable principal **Dr. C. D. Naidu** for this unflinching devotion, which lead us to complete this project. The support, encouragement given by him and his motivation lead us to complete this project.

We express our thanks to internal guide **Mrs.L.Indira** and also Head of the department **Dr. B. V. Kiranmayee** for having provided us a lot of facilities to undertake the project work and guide us to complete the project.

We express our sincere thanks to our faculty of the department of **Computer Science and Engineering** and the remaining members of our college **VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING** **AND TECHNOLOGY** who extended their valuable support in helping us to complete the project in time.

B.Shashidhar(18071A0562)

Ch.Chakrapani(18071A0570)

K.Nikhil(18071A0584)

B.Kranthi(19075A0570)

**ABSTRACT**

Wine industry shows a recent growth spurt as social drinking is on the rise. The price of wine depends on a rather abstract concept of wine appreciation by wine tasters, opinion among whom may have a high degree of variability. Another key factor in wine certification and quality assessment is physicochemical tests which are laboratory-based and takes into account factors like acidity, pH level, the presence of sugar and other chemical properties. For the wine market, it would be of interest if human quality of tasting can be related to the chemical properties of wine so that certification and quality assessment and assurance process is more controlled. Data are collected on 12 different properties of the wines one of which is Quality, based on sensory data, and the rest are on chemical properties of the wines including density, acidity, alcohol content etc. All chemical properties of wines are continuous variables. Quality is an ordinal variable with a possible ranking from 1 (worst) to 10 (best). Each variety of wine is tasted by three independent tasters and the final rank assigned is the median rank given by the taster. Wine classification is a difficult task since taste is the least understood of the human senses. A good wine quality prediction can be very useful in the certification phase, since currently the sensory analysis is performed by human tasters, being clearly a subjective approach. An automatic predictive system can be integrated into a decision support system, helping the speed and quality of the performance. Furthermore, a feature selection process can help to analyze the impact of the analytical tests. If it is concluded that several input variables are highly relevant to predict the wine quality, since in the production process some variables can be controlled, this information can be used to improve the wine quality. Classification models used here are 1) Random Forest 2) SVM 3) Decision Tree

**INDEX**

**Contents**

[**CHAPTER 1 1**](#_Toc49962222)

[**INRODUCTION 1**](#_Toc49962223)

[**1.1Introduction to Machine Learning 1**](#_Toc49962224)

[**1.2 Some machine learning methods 1**](#_Toc49962225)

[**1.3 Machine learning in Wine industry to improve quality 2**](#_Toc49962226)

[**CHAPTER 2 4**](#_Toc49962227)

[**2.1 EXSISTING SYSTEM 4**](#_Toc49962228)

[**2.2 PROPOSED SYSTEM 4**](#_Toc49962229)

[**CHAPTER 3 5**](#_Toc49962230)

[**FEASIBILITY STUDY 5**](#_Toc49962231)

[**CHAPTER 4 6**](#_Toc49962232)

[**SYSTEM ANALYSIS 6**](#_Toc49962233)

[**4.1 SYSTEM REQUIREMENTS 6**](#_Toc49962234)

[**4.1.1 SOFTWARE REQUIREMENTS 6**](#_Toc49962235)

[**4.1.2 HARDWARE REQUIREMENTS 7**](#_Toc49962236)

[**CHAPTER 5 8**](#_Toc49962237)

[**SOFTWARE DESIGN 8**](#_Toc49962238)

[**5.1 Data set preparation 9**](#_Toc49962239)

[**5.2 Splitting the dataset 9**](#_Toc49962240)

[**5.3 Models Description 10**](#_Toc49962241)

[**5.3.1 Decision Trees 10**](#_Toc49962242)

[**5.3.2 Random Forest Model 11**](#_Toc49962243)

[**5.3.3 Support Vector Machine (SVM) 13**](#_Toc49962244)

[**5.4 Training and Testing 15**](#_Toc49962245)

**5.4.1**[**Training Data 15**](#_Toc49962246)

**5.4.2**[**Test Data 15**](#_Toc49962247)

[**5.5 Tuning 17**](#_Toc49962248)

[**5.6 Accuracy, Precision, Recall 17**](#_Toc49962249)

[**CHAPTER 6 18**](#_Toc49962250)

[**IMPLEMENTATION 18**](#_Toc49962251)

[**6.1 Importing python API’s and libraries 18**](#_Toc49962252)

[**6.2 Uploading dataset 19**](#_Toc49962253)

[**6.3 Reading the data by creating a data frame and printing data set 20**](#_Toc49962254)

[**6.4 Searching for NULL values 20**](#_Toc49962255)

[**6.5 Constructing a graph quality v/s count 21**](#_Toc49962256)

[**6.6 Grouping the dataset 22**](#_Toc49962257)

[**6.7 Correlation matrix 23**](#_Toc49962258)

[**6.8 Normalizing the data 26**](#_Toc49962259)

[**CHAPTER 7 27**](#_Toc49962260)

[**TRAINING AND TESTING 27**](#_Toc49962261)

[**7.1 Splitting the data 27**](#_Toc49962262)

[**7.2 Training and testing the Decision Tree model 27**](#_Toc49962263)

[**7.2.1 Training 27**](#_Toc49962264)

[**7.2.2 Testing and accuracy 28**](#_Toc49962265)

[**7.3 Training and testing the Random Forest Model 29**](#_Toc49962266)

[**7.3.1 Training 29**](#_Toc49962267)

[**7.3.2 Testing and finding accuracy 29**](#_Toc49962268)

[**7.4 Training and Testing the SVM (Support Vector Machine) Model. 30**](#_Toc49962269)

[**7.4.1 Training 30**](#_Toc49962270)

[**7.4.2 Testing and finding accuracy 30**](#_Toc49962271)

[**CHAPTER 8 32**](#_Toc49962272)

[**CONCLUSION AND FUTURE WORK 32**](#_Toc49962273)

[**CHAPTER 9 34**](#_Toc49962274)

[**REFERENCES (BIBILOGRAPHY) 34**](#_Toc49962275)

**LIST OF FIGURES**

**Table of Contents page no**

**Fig:4.1 python logo 6**

**Fig:4.2 jupyter notebook7**

**Fig:5.1 Software design model 8**

**Fig:5.2 dataset preparation…………………………………………………………..9**

**Fig 5.3 Decision Tree………………………………………………………………11**

**Fig 5.4 Random Forest……………………………………………………………...12**

**Fig 5.5 SVM………………………………………………………………………….14**

**Fig:5.6 training and testing the model with the dataset…………………………..15**

**Fig 6.1 Importing libraries18**

**Fig:6.2 Uploading dataset19**

**Fig:6.3 Output 20**

**Fig:6.4 reading and printing data………………………………………………….20**

**Fig:6.5 NULL values………………………………………………………………21**

**Fig:6.6 Code for histogram………………………………………………………...21**

**Fig:6.7 Histogram……………………………..…………………………………….22**

**Fig:6.8 grouping……………………………………………………………………..22**

**Fig 6.9 Classification………………………………………………………………23**

**Fig 6.10 Code for correlation matrix…………………………………………….23**

**Fig 6.11 Correlation Matrix………………………………………………………..23**

**Fig 6.12 Printing correlation values………………………………………………..24**

**Fig:6.13 rating v/s alcohol…………………………………………………………..25**

**Fig:6.14 rating v/s sulphates………………………………………………………..25**

**Fig:6.15 rating v/s pH……………………………………………………………….26**

**Fig:6.16 normalization……………………………………………………………...26**

**Fig 7.1: Splitting……………………………………………………………………..27**

**Fig 7.2: Training the decision tree………………………………………………...27**

**Fig 7.3: Testing the decision tree…………………………………………………...28**

**Fig 7.4: Test report………………………………………………………………….28**

**Fig 7.5: Accuracy of decision tree………………………………………………….28**

**Fig 7.6: Training the random forest………………………………………………..29**

**Fig 7.7: Testing the random forest model………………………………………….29**

**Fig 7.8: Test report of random forest………………………………………………29**

**Fig 7.9: Accuracy of random forest model…………...……………………………30**

**Fig 7.10: Training SVM model……………………………………………………..30**

**Fig 7.11: Testing the SVM model…………………………………………..………30**

**Fig 7.12: Test report of SVM model………………………………………………..31**

**Fig 7.13: Accuracy of SVM model………………………………………………….31**

# **CHAPTER 1**

# **INRODUCTION**

# **1.1Introduction to Machine Learning**

Machine Learning is a tender which offers frameworks with capability to perfunctorily take in and enhance from information it secures without really being modified to do as such. This idea accentuates on the advancement of projects that can consequently breakdown information and utilize it to absorb for themselves to improve decisions. This idea is critical to the area of Artificial Intelligence.

Learning initiates with data and impression of this data, for instance, coordinate finding, or direction, or representations, to search for designs, present regularly in material and at last distinguish between poor and enhanced judgments later with respect to cases that we give. The primary goal is to allow the PCs to learn ordinarily without human interpolation and modify exercises as indicated by the required.

ML enables programming applications to wind up more precise in anticipating results without being expressly modified. The essential introduce of ML is to fabricate calculations that can get input information and utilize measurable going-over to get ahead a yield an incentive inside a satisfactory range.

# **1.2 Some machine learning methods**

ML algorithms are categorized as supervised or unsupervised.

1. **Supervised Learning** calculations can remove those that have been recognized previously to fresh information utilizing named cases to foresee future occasions. Beginning since examination of a preparation dataset, the calculation delivers a gathered capacity to make forecasts about the yield esteems. The framework can give efforts to any first-hand contribution after suitable preparing. The learning calculation can likewise contrast its yield and the right, expected income and realize blunders with a specific end objective to adjust the model as needs be.
2. In differentiate, **unsupervised ML** deviousness are developed when the data used to prepare is neither arranged nor marked. Unsupervised learning envisages how frameworks can surmise a capacity to portray a concealed structure from unlabeled information.
3. **Semi-regulated ML** calculations fall some place in the middle of managed and unsupervised learning, since they utilize both unlabeled information as well as marked one for preparing measures of some labelled information and a lot of unlabeled information. The frameworks that utilization this strategy can impressively enhance learning precision. Typically, semi-administered learning is picked when the obtained named information requires gifted and pertinent assets with a specific end goal to prepare it/gain from it.
4. **Active learning** is a machine learning approach that lets users play an active role in the learning process. An active learning approach can ask a user (e.g., a domain expert) to label an example, which may be from a set of unlabeled examples or synthesized by the learning program. The goal is to optimize the model quality by actively acquiring knowledge from human users, given a constraint on how many examples they can be asked to label.

# **1.3 Machine learning in Wine industry to improve quality**

Nowadays wine is increasingly enjoyed by a wider range of consumers. Due to this, the industry is investing in new technologies for both wine making and selling processes. Wine certification and quality assessment are key elements within this context. Wine certification is often assessed by physicochemical and sensory tests. Physicochemical laboratory tests routinely used to characterize wine include determination of density, alcohol or pH values, while sensory tests rely mainly on human experts. It should be stressed that taste is the least understood of the human senses, thus wine classification is a difficult task.

Moreover, the relationships between the physicochemical and sensory analysis are complex and still not fully understood. On the other hand, advances in information technologies have made it possible to collect, store and process massive, often highly complex datasets. All this data hold valuable information such as trends and patterns, which can be used to improve decision making and optimize chances of success. Data mining (DM) techniques aim at extracting high-level knowledge from raw data. There are several DM algorithms, each one with its own advantages. When modelling continuous data, the linear/multiple regression (MR) is the classic approach. Neural networks (NNs) have become increasingly used since the introduction of the backpropagation algorithm. More recently, support vector machines (SVMs) have also been proposed. Due to their higher flexibility and nonlinear learning capabilities, both NNs and SVMs are gaining an attention within the DM field, often attaining high predictive performances. SVMs present theoretical advantages over NNs, such as the absence of local minima in the learning phase.

# **CHAPTER 2**

**EXSISTING SYSTEM AND PROPOSED SYSTEM**

# **2.1 EXSISTING SYSTEM**

The exsisting system deals with the traditional way of assesing the quality.The wine produced are taken as samples and both sensory and physiochemical tests (analytical chemistry) .The quality is finalised based on the results of the physiochemical tests.The results of wine experts and chemist are combined together determine the quality of the wine in the wine industry.

# **2.2 PROPOSED SYSTEM**

* In the exsisting system chemist and wine experts work seperately on the wine equilbrium and senses and results combined together and this consume more time to give the quality.
* And also other resources like money and number of people required to complete the task are high.
* In our system we collect the 11 variables (contents) of wine and feed it into our machine learning algorithm and gets the output variable as quality.
* Our system gives insights of the dependency of target varibles on independent variables using machine learning techniques to determine the quality of wine because it gives the best outcome for the asurance of quality of wine.
* The dependent varible is “quality” where other variables i.e. alcohol,sulphur etc. are assumed to be predictors or independent variables
* We used classsification models like Random Forest,SVM(Support Vector Machine),Decision Tree to train and test our dataset.

# **CHAPTER 3**

# **FEASIBILITY STUDY**

Wine classification is a difficult task since taste is the least understood of the human senses. A good wine quality prediction can be very useful in the certification phase, since currently the sensory analysis is performed by human tasters, being clearly a subjective approach. Quality assessment are crucial issues within the wine industry. AI is a technology that is transforming every walk of life. It is a wide-ranging tool that enables people to rethink how we integrate information, analyze data, and use the resulting insights to improve decision making. Our hope through this comprehensive overview is to explain AI to an audience of policymakers, opinion leaders, and interested observers, and demonstrate how AI already is altering the world and raising important questions for society, the economy, and governance. The idea of using machine learning to in order to improve the wine quality saves the resources of wine quality and it is also easily adaptable in present situation of technological world. The supervised learning done on the contents of the wine gives an idea of quality of the wine with in the less time than compared to the exsisting methods. This process is very cost effective and time saving. The wine produced taken as sample and a quantitative analysis is done on it to get the quantities of the 11 variables (Fixed acidity,Volatile acidity,Citric acid, Residual sugar, Chlorides, Free sulphuric acid, Total sulphuric acid, Density, pH , Sulphates, alcohol) which we use as input to our model and get the output variable “quality” .Further we can study the correlation between the output and input variables and also between input variables.

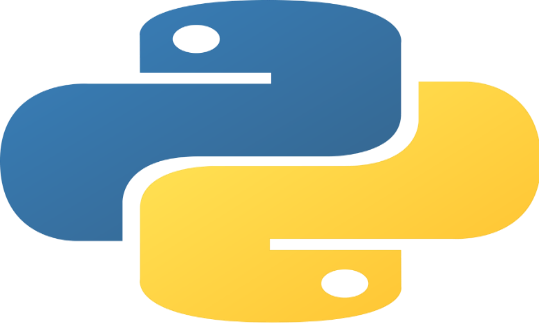
# **CHAPTER 4**

# **SYSTEM ANALYSIS**

## **4.1 SYSTEM REQUIREMENTS**

# **4.1.1 SOFTWARE REQUIREMENTS**

**a) Python**

Python is a deciphered language Guido van Rossum, Python has a diagram hypothesis that complements code decipherability, and a sentence structure that empowers programming architects to express thoughts in less lines of code noticeably using imperative whitespace. It gives builds up that engage little immense. Incorporates a kind modified organization. Reinforces different perfect models, masterminded, essential, useful and, and a huge and exhaustive.

**Fig:4.1 python logo**

**b) Jupyter Notebook**

The Jupyter Note pad is an open-source web application that allows you to make and offer reports that cover live code, conditions, observations and account content. Utilizations include: information cleaning and change, numerical reproduction, factual demonstrating, information representation, machine learning, and significantly more. The Scratch pad has bolster for more than 40 programming dialects, including Python, R, Julia, and Scala. Note pads can be imparted to others using email, Dropbox, GitHub and the Jupyter Note pad Watcher.Colab is a free Jupyter notebook environment that runs entirely in the cloud. Most importantly, it does not require a setup and the notebooks that you create can be simultaneously edited by your team members - just the way you edit documents in Google Docs. Colab supports many popular machine learning libraries which can be easily loaded in your notebook. 

**Fig:4.2 jupyter notebook**

# **4.1.2 HARDWARE REQUIREMENTS**

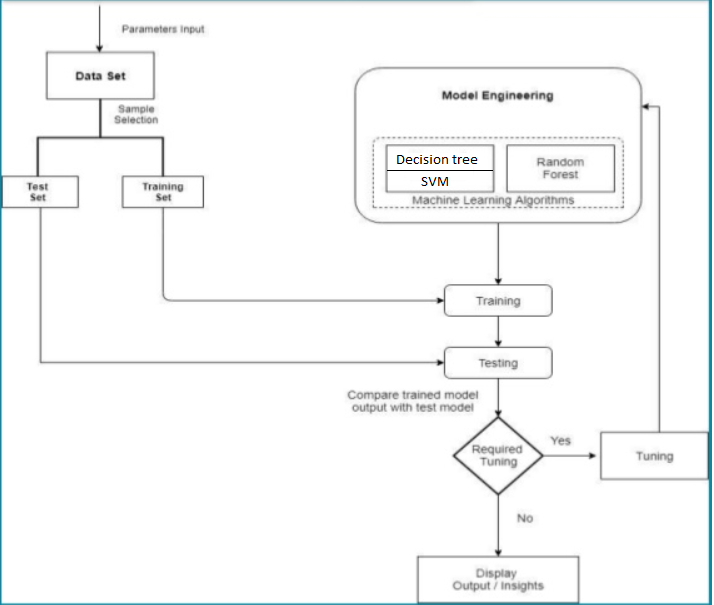
**RAM:** 4GB and higher

**Processor:** Intel i3 and above

**Hard disk:** 10GB minimum

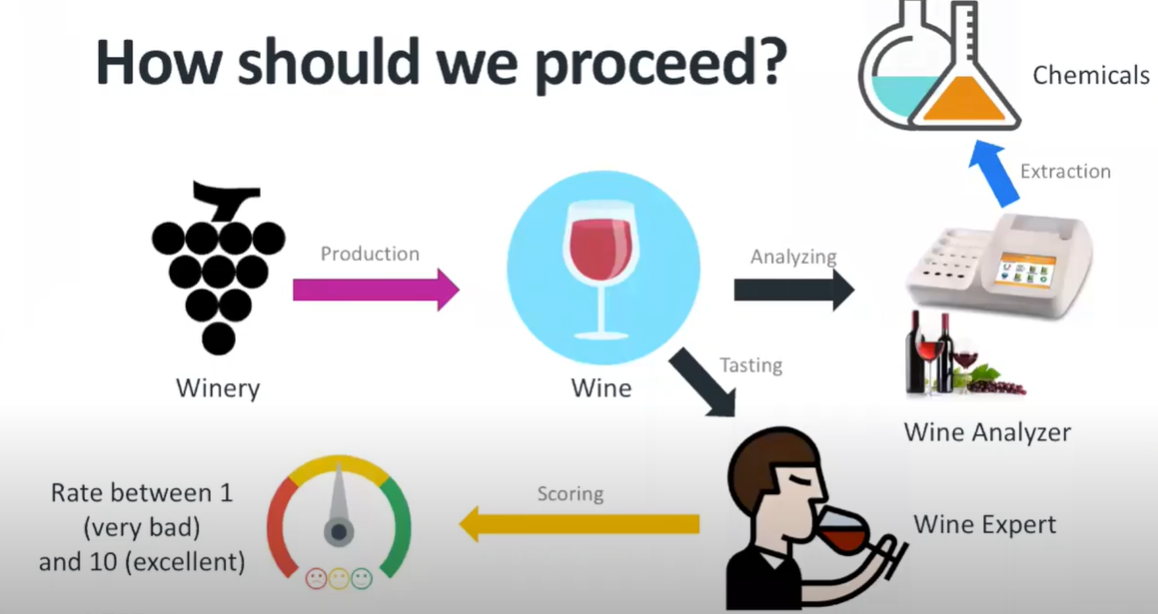
# **CHAPTER 5**

# **SOFTWARE DESIGN**



**Fig:5.1 Software design model**

# **5.1 Data set preparation**



**Fig 5.2:data set preparation**

The wine is produced in the industry by the process of fermentation in which the yeast consumes the sugar and produces the alcohol and CO2 as the products. After 20-30 days the wine is produced by mixing the additives in between the process. Now the wine sample is analysed by the extracting the chemicals quantities and studying the chemical results and the sensory results from wine experts the wine is given a score for its quality .So this process is repeated for 1599 samples and the dataset is prepared as 12 columns consisting of Fixed acidity,Volatile acidity,Citric acid, Residual sugar, Chlorides, Free sulphuric acid, Total sulphuric acid, Density, pH , Sulphates, alcohol and quality.

# **5.2 Splitting the dataset**

Now the prepared dataset is used for our machine learning model.The train-test split is a technique for evaluating the performance of a machine learning algorithm. It can be used for classification or regression problems and can be used for any supervised learning algorithm. The procedure involves taking a dataset and dividing it into two subsets. The first subset is used to fit the model and is referred to as the training dataset. The second subset is not used to train the model; instead, the input element of the dataset is provided to the model, then predictions are made and compared to the expected values. This second dataset is referred to as the test dataset.

* Train Dataset: Used to fit the machine learning model.
* Test Dataset: Used to evaluate the fit machine learning model

The objective is to estimate the performance of the machine learning model on new data: data not used to train the model. This is how we expect to use the model in practice. Namely, to fit it on available data with known inputs and outputs, then make predictions on new examples in the future where we do not have the expected output or target values. The train-test procedure is appropriate when there is a sufficiently large dataset available.

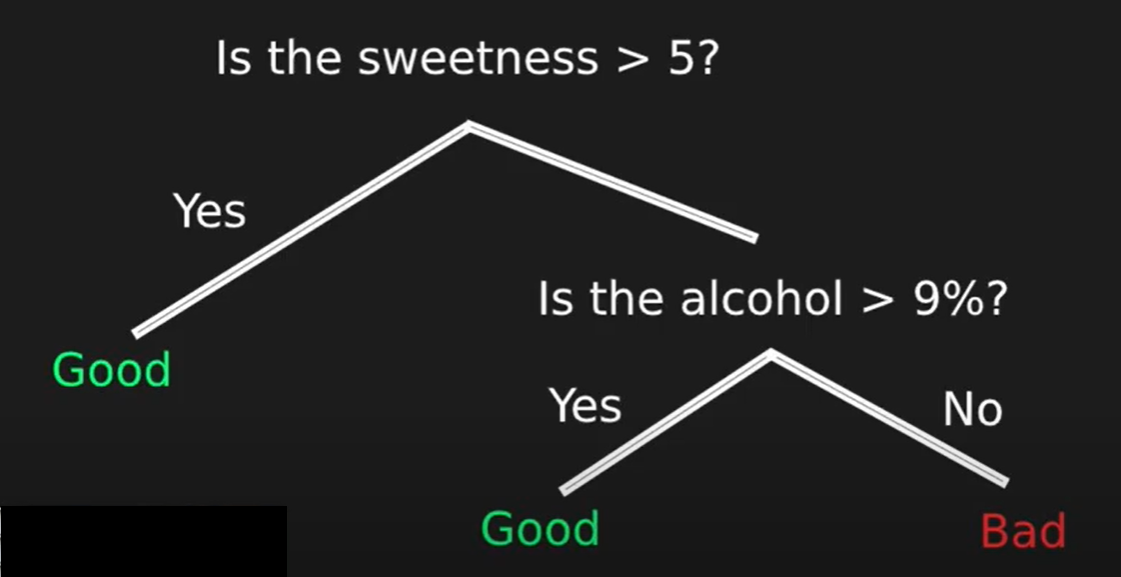
# **5.3 Models Description**

The Machine learning models used in our project are

* Decision Trees
* Random Forest
* Support Vector Machine (SVM)

# **5.3.1 Decision Trees**

Decision tree learning is one of the predictive modelling approaches used in [statistics](https://en.wikipedia.org/wiki/Statistics), [data mining](https://en.wikipedia.org/wiki/Data_mining) and [machine learning](https://en.wikipedia.org/wiki/Machine_learning). It uses a [decision tree](https://en.wikipedia.org/wiki/Decision_tree) (as a [predictive model](https://en.wikipedia.org/wiki/Predictive_modelling)) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, [leaves](https://en.wikipedia.org/wiki/Leaf_node) represent class labels and branches represent [conjunctions](https://en.wikipedia.org/wiki/Logical_conjunction) of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically [real numbers](https://en.wikipedia.org/wiki/Real_numbers)) are called regression trees. Decision trees are among the most popular machine learning algorithms given their intelligibility and simplicity.



**Fig 5.3 Decision Tree**

**Advantages**

* Simple to understand and interpret
* Able to handle both numerical and [categorical](https://en.wikipedia.org/wiki/Categorical_variable) data
* Requires little data preparation
* Uses a [white box](https://en.wikipedia.org/wiki/White_box_(software_engineering)) or open-box[[2]](https://en.wikipedia.org/wiki/Decision_tree_learning#cite_note-:1-2) model
* Possible to validate a model using statistical tests
* Performs well with large datasets
* Mirrors human decision making more closely than other approaches

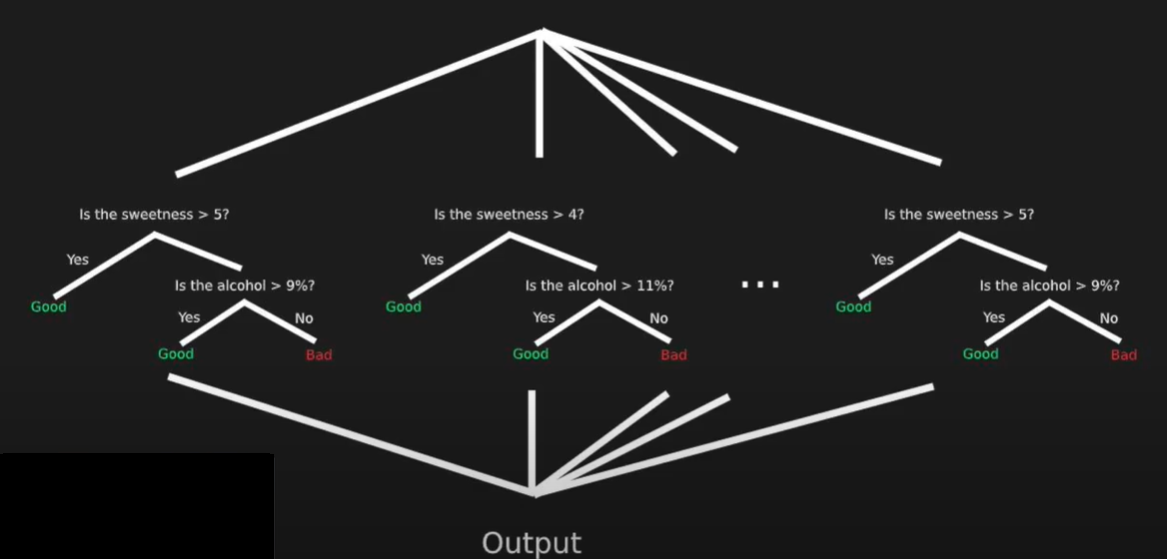
**Limitations**

* Trees can be very non-robust. A small change in the [training data](https://en.wikipedia.org/wiki/Training,_test,_and_validation_sets) can result in a large change in the tree and consequently the final predictions
* Decision-tree learners can create over-complex trees that do not generalize well from the training data. (This is known as [overfitting](https://en.wikipedia.org/wiki/Overfitting).)
* The problem of learning an optimal decision tree is known to be [NP-complete](https://en.wikipedia.org/wiki/NP-complete) under several aspects of optimality and even for simple concepts

# **5.3.2 Random Forest Model**

Random forests or random decision forests are an [ensemble learning](https://en.wikipedia.org/wiki/Ensemble_learning) method for [classification](https://en.wikipedia.org/wiki/Statistical_classification), [regression](https://en.wikipedia.org/wiki/Regression_analysis) and other tasks that operate by constructing a multitude of [decision trees](https://en.wikipedia.org/wiki/Decision_tree_learning) at training time and outputting the class that is the [mode](https://en.wikipedia.org/wiki/Mode_(statistics)) of the classes (classification) or mean prediction (regression) of the individual trees.[[1]](https://en.wikipedia.org/wiki/Random_forest#cite_note-ho1995-1)[[2]](https://en.wikipedia.org/wiki/Random_forest#cite_note-ho1998-2) Random decision forests correct for decision trees' habit of [overfitting](https://en.wikipedia.org/wiki/Overfitting) to their [training set](https://en.wikipedia.org/wiki/Test_set).

A decision tree is built on an entire dataset, using all the features/variables of interest, whereas a random forest randomly selects observations/rows and specific features/variables to build multiple decision trees from and then averages the results.



**Fig 5.4: Random Forest**

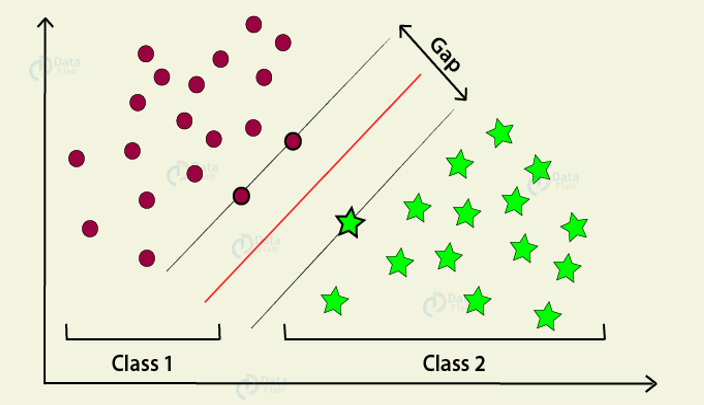
**Advantages**

1. Random Forest is based on the bagging algorithm and uses Ensemble Learning technique. It creates as many trees on the subset of the data and combines the output of all the trees. In this way it reduces overfitting problem in decision trees and also reduces the variance and therefore improves the accuracy.  
2. Random Forest can be used to solve both classification as well as regression problems.  
3. Random Forest works well with both categorical and continuous variables.  
4. Random Forest can automatically handle missing values.  
5. No feature scaling required: No feature scaling (standardization and normalization) required in case of Random Forest as it uses rule based approach instead of distance calculation.  
6. Handles non-linear parameters efficiently: Non linear parameters don't affect the performance of a Random Forest unlike curve based algorithms. So, if there is high non-linearity between the independent variables, Random Forest may outperform as compared to other curve based algorithms.  
7. Random Forest can automatically handle missing values.  
8. Random Forest is usually robust to outliers and can handle them automatically.  
9. Random Forest algorithm is very stable. Even if a new data point is introduced in the dataset, the overall algorithm is not affected much since the new data may impact one tree, but it is very hard for it to impact all the trees.  
10. Random Forest is comparatively less impacted by noise.  
  
**Disadvantages of Random Forest**  
1. **Complexity**: Random Forest creates a lot of trees (unlike only one tree in case of decision tree) and combines their outputs. By default, it creates 100 trees in Python sklearn library. To do so, this algorithm requires much more computational power and resources. On the other hand decision tree is simple and does not require so much computational resources.  
2. **Longer Training Period**: Random Forest require much more time to train as compared to decision trees as it generates a lot of trees (instead of one tree in case of decision tree) and makes decision on the majority of votes.

# **5.3.3 Support Vector Machine (SVM)**

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N-the number of features) that distinctly classifies the data points. To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier. Deleting the support vectors will change the position of the hyperplane. These are the points that help us build our SVM.



**Fig 5.5 SVM**

**Advantages**

* SVM works relatively well when there is clear margin of separation between classes.
* SVM is more effective in high dimensional spaces.
* SVM is effective in cases where number of dimensions is greater than the number of samples.
* SVM is relatively memory efficient.

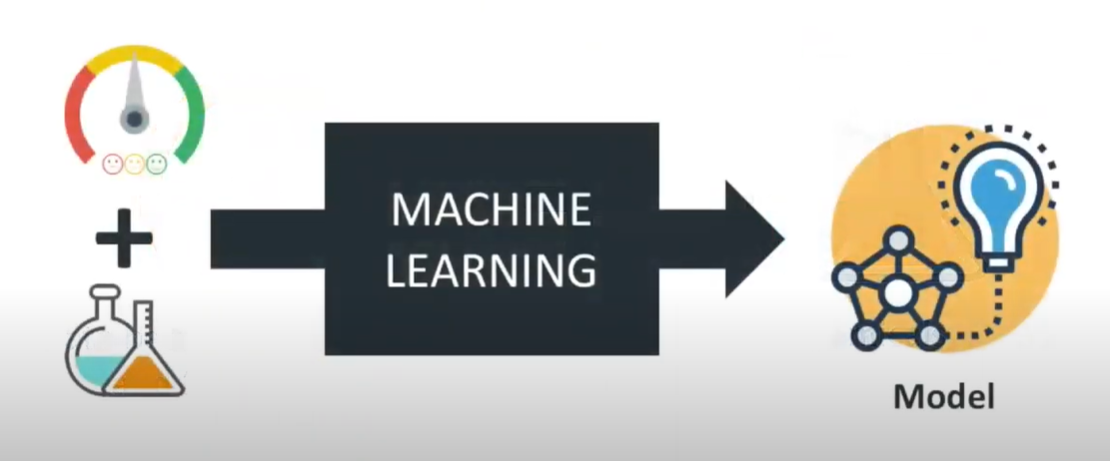
**Disadvantages**

* SVM algorithm is not suitable for large data sets.
* SVM does not perform very well, when the data set has more noise i.e. target classes are overlapping.
* In cases where number of features for each data point exceeds the number of training data sample, the SVM will under perform.
* As the support vector classifier works by putting data points, above and below the classifying hyper plane there is no probabilistic explanation for the classification.

# **5.4 Training and Testing**

**Training Data**

The observations in the training set form the experience that the algorithm uses to learn. In supervised learning problems, each observation consists of an observed output variable and one or more observed input variables.



**Fig:5.6 training and testing the model with the dataset**

**Test Data**

The test set is a set of observations used to evaluate the performance of the model using some performance metric. It is important that no observations from the training set are included in the test set. If the test set does contain examples from the training set, it will be difficult to assess whether the algorithm has learned to generalize from the training set or has simply memorized it.

A program that generalizes well will be able to effectively perform a task with new data. In contrast, a program that memorizes the training data by learning an overly complex model could predict the values of the response variable for the training set accurately, but will fail to predict the value of the response variable for new examples. Memorizing the training set is called **over-fitting**. A program that memorizes its observations may not perform its task well, as it could memorize relations and structures that are noise or coincidence. Balancing memorization and generalization, or over-fitting and under-fitting, is a problem common to many machine learning algorithms. **Regularization** may be applied to many models to reduce over-fitting.

In addition to the training and test data, a third set of observations, called a **validation** or **hold-out set**, is sometimes required. The validation set is used to tune variables called **hyper parameters**, which control how the model is learned. The program is still evaluated on the test set to provide an estimate of its performance in the real world; its performance on the validation set should not be used as an estimate of the model's real-world performance since the program has been tuned specifically to the validation data. It is common to partition a single set of supervised observations into training, validation, and test sets. There are no requirements for the sizes of the partitions, and they may vary according to the amount of data available. It is common to allocate 50 percent or more of the data to the training set, 25 percent to the test set, and the remainder to the validation set.

Some training sets may contain only a few hundred observations; others may include millions. Inexpensive storage, increased network connectivity, the ubiquity of sensor-packed smartphones, and shifting attitudes towards privacy have contributed to the contemporary state of big data, or training sets with millions or billions of examples.

However, machine learning algorithms also follow the maxim "garbage in, garbage out." A student who studies for a test by reading a large, confusing textbook that contains many errors will likely not score better than a student who reads a short but well-written textbook. Similarly, an algorithm trained on a large collection of noisy, irrelevant, or incorrectly labeled data will not perform better than an algorithm trained on a smaller set of data that is more representative of problems in the real world.

Many supervised training sets are prepared manually, or by semi-automated processes. Creating a large collection of supervised data can be costly in some domains. Fortunately, several datasets are bundled with **scikit-learn**, allowing developers to focus on experimenting with models instead.

During development, and particularly when training data is scarce, a practice called **cross-validation** can be used to train and validate an algorithm on the same data. In cross-validation, the training data is partitioned. The algorithm is trained using all but one of the partitions, and tested on the remaining partition. The partitions are then rotated several times so that the algorithm is trained and evaluated on all of the data.

# **5.5 Tuning**

Tuning is usually a trial-and-error process by which you change some hyperparameters (for example, the number of trees in a tree-based algorithm or the value of alpha in a linear algorithm), run the algorithm on the data again, then compare its performance on your validation set in order to determine which set of hyperparameters results in the most accurate model.

All [machine learning](https://www.datarobot.com/wiki/machine-learning/) [algorithms](https://www.datarobot.com/wiki/algorithm/) have a “default” set of hyperparameters, which [Machine Learning Mastery](https://machinelearningmastery.com/difference-between-a-parameter-and-a-hyperparameter/) defines as “a configuration that is external to the [model](https://www.datarobot.com/wiki/model/) and whose value cannot be estimated from data.” Different algorithms consist of different hyperparameters. For example, regularized [regression](https://www.datarobot.com/wiki/regression/) models have coefficients penalties, decision trees have a set number of branches, and [neural networks](https://www.datarobot.com/wiki/neural-network/) have a set number of layers. When building models, analysts and data scientists choose the default configuration of these hyperparameters after running the model on several datasets.

While the generic set of hyperparameters for each algorithm provides a starting point for analysis and will generally result in a well-performing model, it may not have the optimal configurations for your particular dataset and business problem. In order to find the best hyperparameters for your data, you need to tune them.

# **5.6 Accuracy, Precision, Recall**

Accuracy is calculated with the following formula −

**ACC = (TP + TN)/ (TP + TN + FP + FN)**

Where, TP is the number of true positives

TN is the number of true negatives

FP is the number of false positives

FN is the number of false negatives.

**Precision** is the fraction of the tumors that were predicted to be malignant that are actually malignant. Precision is calculated with the following formula −

**PREC = TP/ (TP + FP)**

**Recall** is the fraction of malignant tumors that the system identified. Recall is calculated with the following formula −

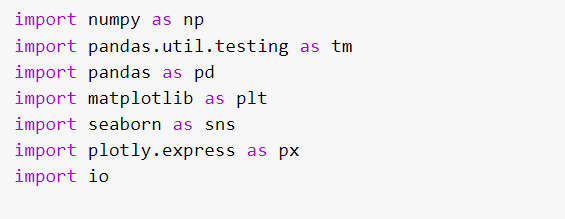
**R = TP/ (TP + FN)**

Training data and test data are two important concepts in machine learning. This chapter discusses them in detail.

# **CHAPTER 6**

# **IMPLEMENTATION**

# **6.1 Importing python API’s and libraries**



**Fig 6.1 Importing libraries**

* **Numpy:**

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation etc

* **Pandas:**

Pandas is a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labelled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real world data analysis in Python. Additionally, it has the broader goal of becoming the most powerful and flexible open source data analysis / manipulation tool available in any language.

* **matplotlib:**

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatterplots, etc., with just a few lines of code.

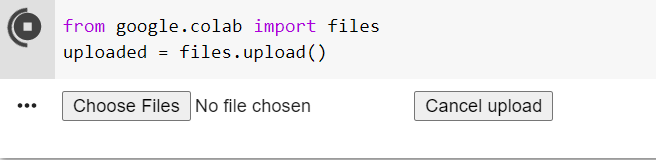
* **Seaborn**

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Seaborn aims to make visualization a central part of exploring and understanding data. Its dataset- oriented plotting functions operate on data frames and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots.

* **sklearn**

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. The library is built upon the SciPy.

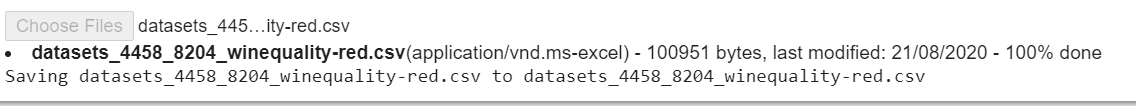
# **6.2 Uploading dataset**



**Fig:6.2 Uploading dataset**

The above gives us two option one is to choose file from our system and other is cancel upload

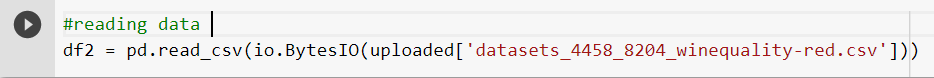
Output:

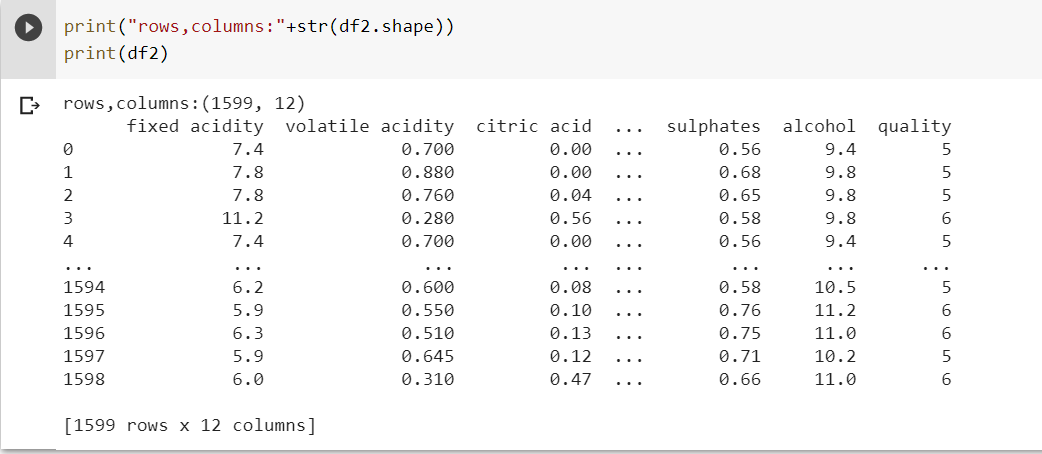


**Fig:6.3 Output**

The dataset is now uploaded successfully

# **6.3 Reading the data by creating a data frame and printing data set**



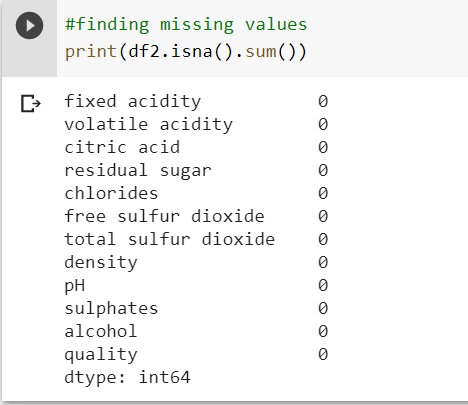


**Fig:6.4 reading and printing dataset**

Therefore, the dataset consists of 1599 rows and 12 columns.

# **6.4 Searching for NULL values**

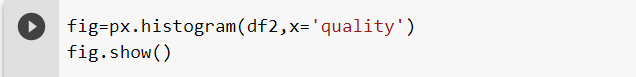
Removing null values from the dataset is one of the important steps in data wrangling. These null values adversely affect the performance and accuracy of any machine learning algorithm. So, it is very important to remove null values from the dataset before applying any machine learning algorithm to that dataset. Although some algorithms like XGBoost have built-in feature to handle null values, but we should also do it manually as a good practice while preparing the data.



**Fig:6.5 NULL values**

So there are no NULL values in the dataset.

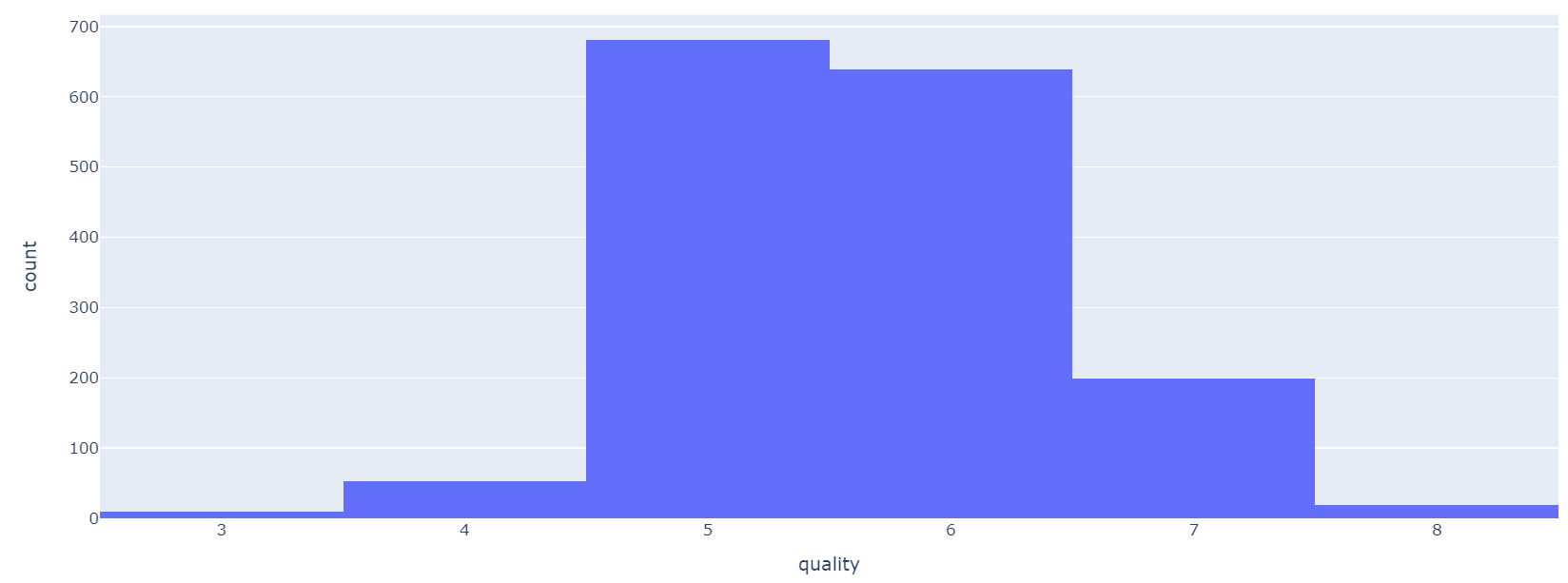
# **6.5 Constructing a graph quality v/s count**



**Fig:6.6 code for histogram**

The above code prints a histogram with quality on x-axis and count on y- axis.

Output:



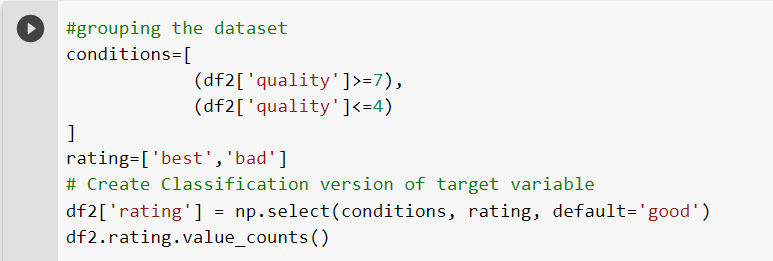
**Fig:6.7 Histogram**

quality is ranging between 1-10 in the dataset and there are more of 5 values in the dataset from the graph

# **6.6 Grouping the dataset**

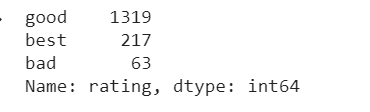
As the quality ranging between 1-10, now we divide the entire dataset into 3 types

* best (quality>=7)
* bad (quality<=4)
* good (4<quality<7)



**Fig:6.8 grouping**

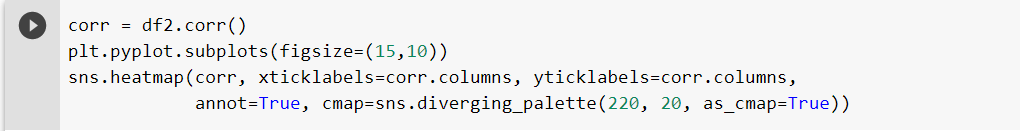
A new column is added to the data frame named rating and the contents of rating are best,good,bad.



**Fig 6.9 classification**

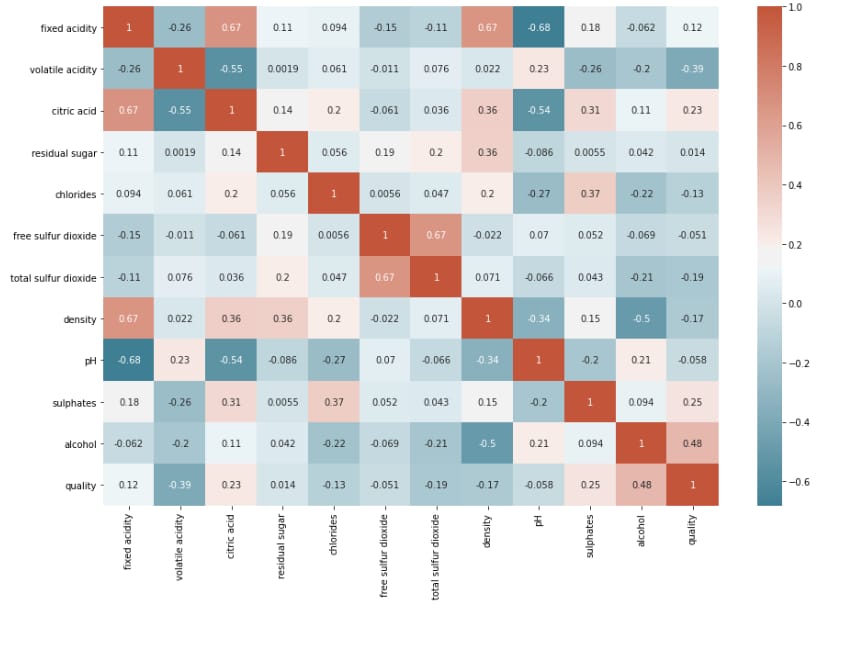
After grouping there are 1319 good quality wine, 217 best quality wine ,63 bad quality wine.

# **6.7 Correlation matrix**

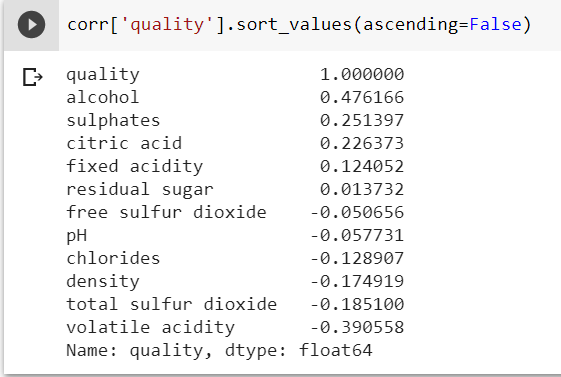


**Fig 6.10 code for correlation matrix**

Output:



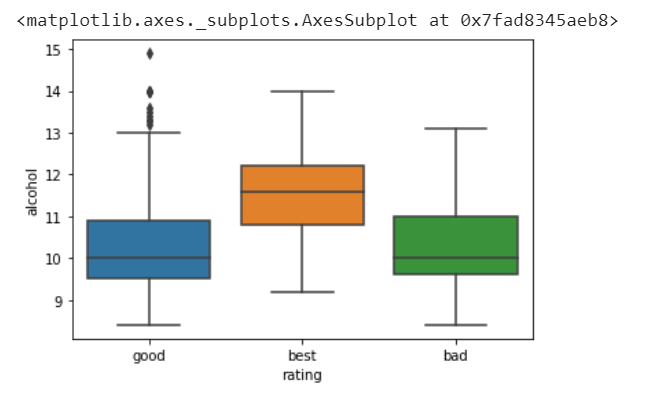
**Fig 6.11 Correlation Matrix**



**Fig 6.12 Printing correlation values**

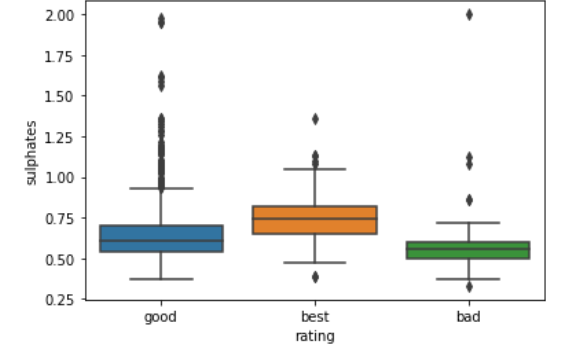
From the above correlation matrix the quality of the wine is positively affected by alcohol, sulphates etc. and negatively affected by pH

So, we construct the graphs between the different variables v/s rating and record our observations.

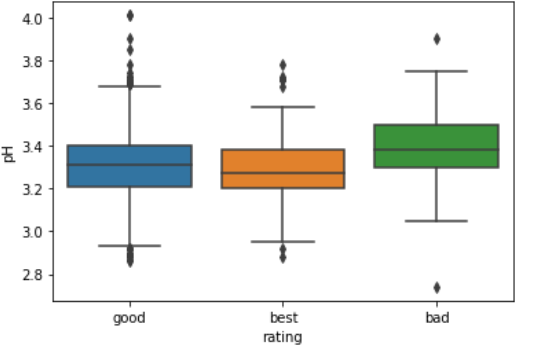


**Fig:6.13 rating v/s alcohol**

* From the graph alcohol content is high in the best rated wine.



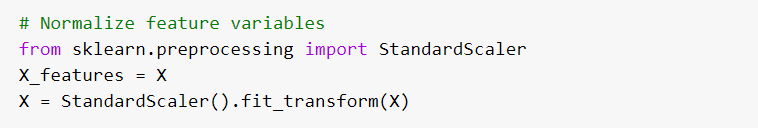
**Fig:6.14 rating v/s sulphates**



**Fig:6.15 rating v/s pH**

* From the graph the bad rated wine has highest pH values.

# **6.8 Normalizing the data**



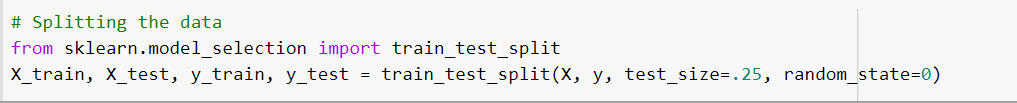
**Fig:6.16 normalization**

 The goal of normalization is to change the values of numeric columns in the dataset to a common scale, without distorting differences in the ranges of values. So, we normalize the data to bring all the variables to the same range.

# **CHAPTER 7**

# **TRAINING AND TESTING**

# **7.1 Splitting the data**



**Fig 7.1: Splitting**

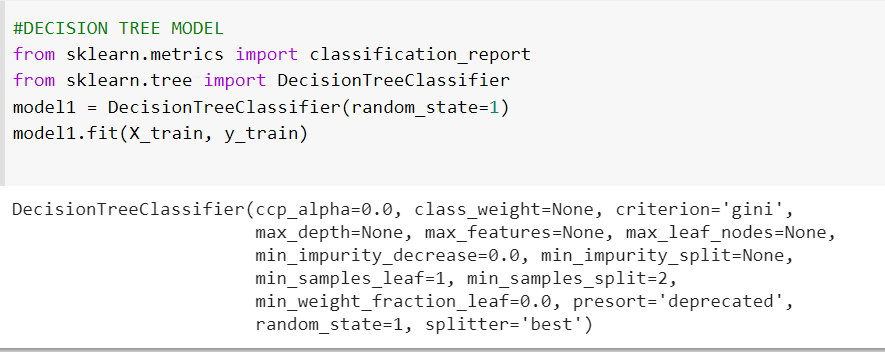
The dataset is splitted into two sets

* Training dataset
* Testing dataset

In splitting the dataset we preserve the 25% of data for the testing after training the 75% of the dataset

# **7.2 Training and testing the Decision Tree model**

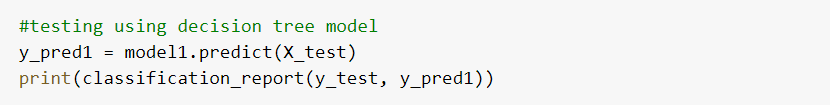
# **7.2.1 Training**



**FIG 7.2: Training the decision tree**

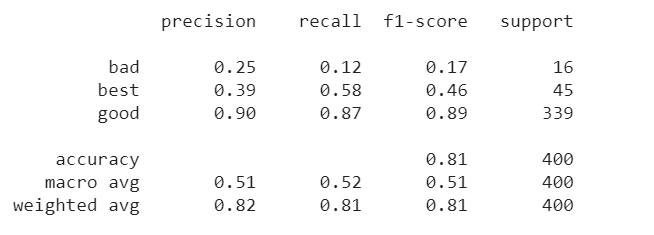
The 75% of the dataset is used to train the decision tree model and after the training the testing is done on the remaining 25% of dataset.

# **7.2.2 Testing and accuracy**

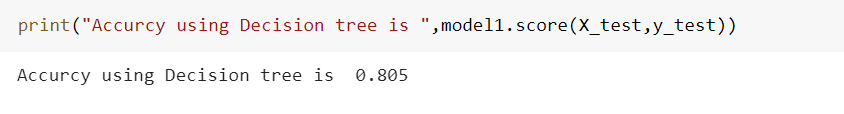


**Fig 7.3: Testing the decision tree**

Taking the X\_test as the independent variable the testing is done (y=f(x)) on the decision tree model



**Fig 7.4: Test report**

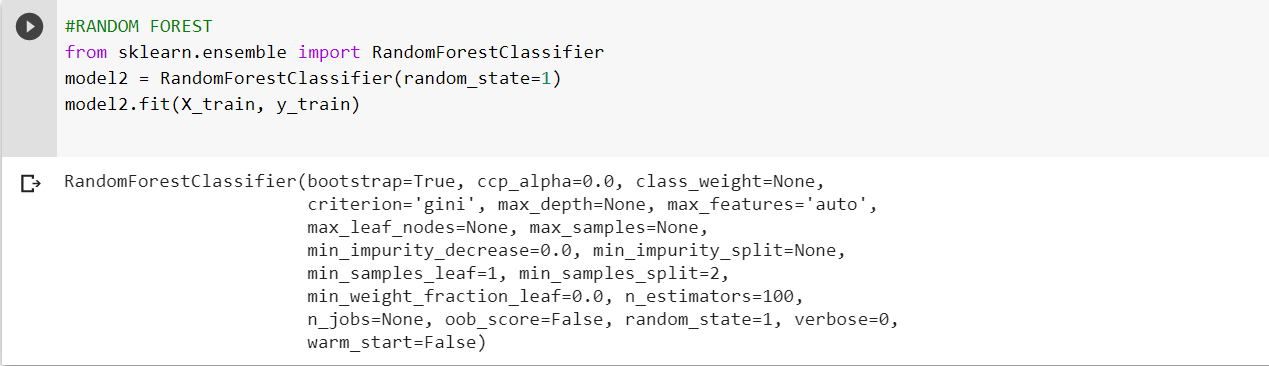


**Fig 7.5: Accuracy of decision tree**

The **Accuracy** of the Decision tree model is **80.5%**.

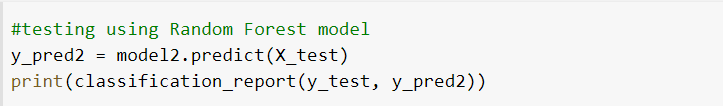
# **7.3 Training and testing the Random Forest Model**

# **7.3.1 Training**

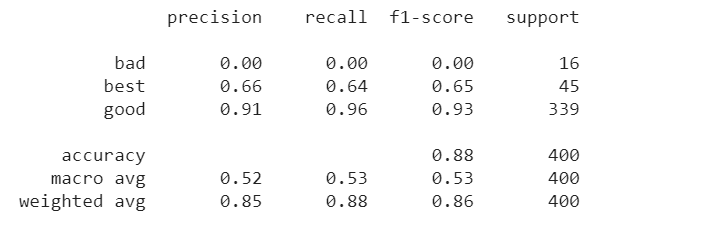


**Fig 7.6: Training the random forest**

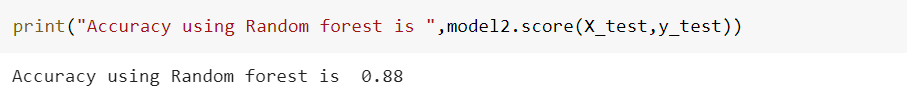
# **7.3.2 Testing and finding accuracy**



**Fig 7.7: Testing the random forest model**



**Fig 7.8: Test report of random forest**

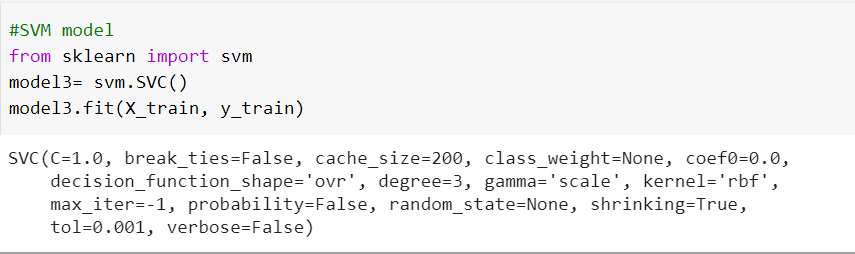


**Fig 7.9: Accuracy of random forest model**

The **Accuracy** of Random Forest Model is **88.0%**.

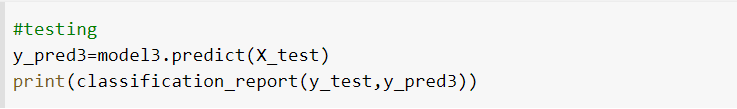
# **7.4 Training and Testing the SVM (Support Vector Machine) Model.**

# **7.4.1 Training**

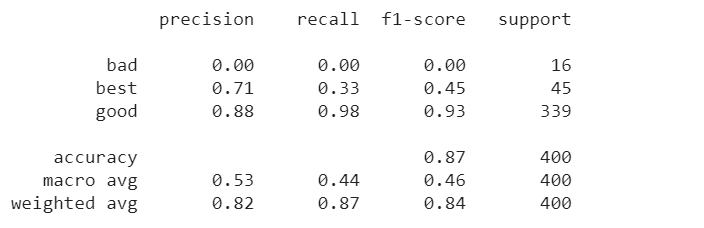


**Fig 7.10: Training SVM model**

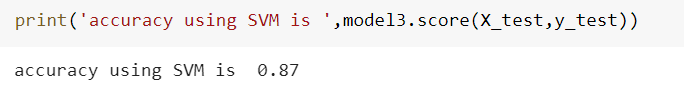
# **7.4.2 Testing and finding accuracy**



**Fig 7.11: Testing the SVM model**



**Fig 7.12: Test report of SVM model**



**Fig 7.13: Accuracy of SVM model**

The **Accuracy** of SVM (Support Vector Machine) is **87.0%.**

# **CHAPTER 8**

# **CONCLUSION AND FUTURE WORK**

Due to the increase in the interest in wine, companies are investing in new technologies to improve their production and selling processes. Quality certification is a crucial step for both processes and is currently dependent on wine tasting by human experts. This work aims at the prediction of wine preferences from objective analytical tests that are available at the certification step. A large dataset was considered. This project was addressed by a classification tasks, where wine preference is modelled in a continuous scale, from 0 -10. This approach preserves the order of the classes, allowing the evaluation of distinct accuracies and also provided a clear idea about the importance of the attributes for prediction of quality, which was time consuming and expensive when done in the traditional way. Due to advances in the Machine learning field, it is possible to extract knowledge from structured data. Indeed, powerful techniques such as neural networks (NNs) and more recently support vector machines (SVMs) are emerging. While being more flexible models (i.e. no a priori restriction is imposed), the performance depends on a correct setting of hyperparameters (e.g. SVM kernel parameter) and the input variables used by the model. Sensitivity analysis is used to extract knowledge from the NN/SVM models, given in terms of the effect on the responses when one input is varied, leading to the proposed Variable Effect Characteristic (VEC) curves, and relative importance of the inputs (measured by the variance of the response changes). Based on the bar plots plotted we come to a conclusion that not all input features are essential and affect the data, for example from the bar plot against quality and residual sugar we see that as the quality increases residual sugar is moderate and does not have change drastically. So, this feature is not so essential as compared to others like alcohol and citric acid, so we can drop this feature while feature selection. For classifying the wine quality, we have implemented multiple algorithms, namely 1) Decision Tree Model 2) Random Forest Model 3) (SVM) Support Vector Machine Model. We were able to achieve maximum accuracy using Random Forest of 88%. Support Vector Machine model giving an accuracy of 87%. Decision Tree has an accuracy of 80.5%.

This model could also be used to improve the training of **oenology** (science and study of wine and winemaking)students. Furthermore, the relative importance of the inputs brought interesting insights regarding the impact of the analytical tests. Since some variables can be controlled in the production process this information can be used to improve the wine quality. For instance, alcohol concentration can be increased or decreased by monitoring the grape sugar concentration prior to the harvest. Also, the residual sugar in wine could be raised by suspending the sugar fermentation carried out by yeasts. Future improvement can be made if more data can be collected on both low-quality and high-quality wine. If the data set has more records on both the low end and high end, the quality of analysis can be improved.

# **CHAPTER 9**

# **REFERENCES (BIBILOGRAPHY)**

* <https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/>
* <https://towardsdatascience.com/predicting-wine-quality-with-several-classification-techniques-179038ea6434#30ac>
* <https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_training_test_data.htm>
* <https://machinelearningmastery.com/train-test-split-for-evaluating-machine-learning-algorithms/#:~:text=The%20reason%20is%20that%20when,effectively%20evaluate%20the%20model%20performance.>
* <https://youtu.be/fwY9Qv96DJY>
* <https://youtu.be/fSytzGwwBVw>
* <https://youtu.be/Kdsp6soqA7o>
* <https://youtu.be/EuBBz3bI-aA>
* <https://www.slideshare.net/MahimaSaini2/wine-quality-analysis-using-machine-learning>
* <https://www.researchgate.net/publication/322347075_Selection_of_important_features_and_predicting_wine_quality_using_machine_learning_techniques>
* <http://theprofessionalspoint.blogspot.com/2019/02/advantages-and-disadvantages-of-random.html>
* <https://en.wikipedia.org/wiki/Random_forest>
* <https://en.wikipedia.org/wiki/Decision_tree>
* <https://youtu.be/2XZ1Qm1OEz4>
* <https://youtu.be/jwiT61vvPlc>