INTERDISCIPLINARY PROJECT REPORT

at

Sathyabama Institute of Science and Technology

(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering Degree in Computer Science and Engineering

By

PASHAM VYSHNAVI

(Reg. No. 40111456)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC | 12 B Status
by UGC | Approved by AICTE
JEPPIAR NAGAR, RAJIV GANDHISALAI,
CHENNAI – 600119

APRIL 2023

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work **PASHAM VYSHNAVI (Reg. No: 40111456)** who carried out the project "**WINE QUALITY PREDICTION**" under mysupervision from Jan 2023 to march 2023

Internal Guide

Dr.T. JUDGI., M.E.,Ph.D.,

Head of the Department

Dr. L. LAKSHMANAN, M.E., Ph.D.,

Submitted for Viva voce Examination held on ____________________

Internal Examiner **External Examiner**

DECLARATION

I **PASHAM VYSHNAVI** hereby declare that the project report entitled "**WINE QUALITY PREDICTION BASED ON MACHINE LEARNING"** done by me under the guidance of **Dr.T.Judgi,M.E.,Ph.D** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering.

DATE:

PLACE: Chennai **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D.**, **Dean**, School of Computing, **Dr. L. Lakshmanan, M.E., Ph.D., Heads of the Department** of **Computer Science and Engineering** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.T.Judgi., M.E.,Ph.D.,** for his valuable guidance, suggestions and constant encouragement paved way for thesuccessful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project

TRAINING CERTIFICATE

NPTEL Online Certification

(Funded by the MoE, Govt. of India)

This certificate is awarded to

VYSHNAVI PASHAM

for successfully completing the course

Machine Learning

with a consolidated score of **55** %

Online Assignments	24.4/25	Proctored Exam	31.06/75

Total number of candidates certified in this course: **583**

Devendra Jalihal

Prof. Devendra Jalihal
Chairperson,
Centre for Outreach and Digital Education, IITM

Jan-Mar 2023

(8 week course)

Prof. Andrew Thangaraj
NPTEL, Coordinator
IIT Madras

Indian Institute of Technology Madras

FREE ONLINE EDUCATION swayam शिक्षित भारत, उन्नत भारत

Roll No: NPTEL23CS11S34337507

To validate the certificate

No. of credits recommended: 2 or 3

ABSTRACT

Wine is an alcoholic drink typically made from fermented grapes. Yeast consumes the sugar in the grapes and converts it into ethanol and carbon dioxide, releasing heat in the process. Different varieties of grapes and strains of yeasts and the ingredients involved are major factors resulting in different types of wine. Also, it' is the most commonly used beverage globally, and its values are considered important in society. Nowadays, industry players are usingproduct quality certifications to promote their products. The Quality of a Wine is important for the consumers as well as the wine industry. The traditional (expert) way of measuring wine quality istime-consuming. Nowadays, machine learning models are important tools to replace human tasks. Our main objective is to predict the wine quality using machine learning through Python programming language. A dataset is considered and wine quality is modeled to analyze the quality of wine through different parameters like fixed acidity, volatile acidity, etc. All these parameters will be analyzed through Machine Learning algorithms like linear regression algorithm which will help to rate the wine on scale 1 - 10 orbad - good. The output obtained would further be checked for correctness and the model will be optimized accordingly. It can support the wine expertevaluations and ultimately improve productivity.

+TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

ABBREVIATION	EXPANSION
ML	MACHINE LEARNING
MAE	MEAN ABSOLUTE ERROR
MSE	MEAN SQUARE ERROR
LR	LINEAR REGRESSION
RMSE	ROOT MEAN SQUARE ERROR

CHAPTER 1

INTRODUCTION

1.1 ABOUT WINE

Wine (derived from Latin vinum) is an alcoholic beverage made from fermented grapes without the addition of sugar, acids, enzymes, water, or other nutrients. Yeast consumes the sugars in the grapes and converts them to ethanol and carbon dioxide. Different grape varieties and yeast types produce different styles of wine. These changes result from the complex interplay between the biochemical development of the grape, fermentation reactions, terroir, and production processes. Non-grape wines include rice wines and fruit wines such as plum, that pomegranate, te, and elderberry. Wine has been produced for thousands of years. Wine quality assessment is one of the key factors in this context and this assessment can be used for certification. This type of quality certification helps ensure the quality of wines on the market. Wine characteristics determine the quality of the wine. In recent years, the availability of many brands of wine has made it difficult to identify good wines. A good wine depends on so many important factors, including chemical, scientific and technical factors Most machine learning techniques can provide highly accurate results that compel most data scientists to implement them when it comes to predictive analytics. Only red and white wine analyzes were considered here. Red wine is made from dark-colored grape varieties. The actual wine color varies from the purple typical of young wines to the red of mature wines to the brown of old red wines. Most purple grape juices are actually greenish-white. The red color comes from anthocyanin pigments (also called anthocyanins) present ie grape skins. The exception is the relatively rare Tenterer variety, which actually has red flesh and produces red juice. White wine is produced by fermenting undyed grape pulp. The grapes used for white wine are usually green or yellow. Other white wines are blended from multiple varietals. Examples are Tokay and Sauternes. A major challenge when analyzing wine variety and quality is ultimately achieving perfect precision in a short amount of time. This can be achieved using server machine-learning techniques.

1.1.1 WINE ATTRIBUTES AND PROPERTIES

Different attributes are present in the wine which determine the wine types and their quality. property properties attributes we have considered are:

• **Fixed acidity:** Acids are one of the fundamental properties of wine and contribute greatly to the taste of the wine. Reducing acids significantly might lead to wines tasting flat. This variable is usually expressed in g(tartaric acid)/dm3 in the dataset.

• **Volatile acidity:** These acids are to be distilled out from the wine before completing the production process. It is primarily constituted of acetic acid. The volatile acidity is expressed in g(acetic acid)/dm3 in the dataset.

• **Citric acid:** This is one of the fixed acids which gives a wine its freshness. Usually, most of it is consumed during the fermentation process, and sometimes it is added separately to give the wine more freshness. It's usuallyexpressed in g/dm3 in the dataset.

• **Residual sugar:** This typically refers to the natural sugar from grapes thatremains after the fermentation process stops, or is stopped. It's usually expressed in g/dm3 in the dataset.

• **Chlorides:** This is usually a major contributor to saltiness in wine. It's usually expressed in g(sodium chloride)/dm3 in the dataset.

• **Free sulfur dioxide:** This is the part of the sulfur dioxide that when added to a bottle of wine is said to be free after the remaining part binds. Winemakers will always try to get the highest proportion of free sulfur to bind. This variable is expressed in mg/dm3 in the dataset.

. Total sulfur dioxide: This is the sum total of the bound and the free sulfurdioxide (SO2). Here, it's expressed in mg/dm3…

. Density: This can be represented as a comparison of the weight of the specific volume of wine to an equivalent volume of water. It is generally used as a measure of t version of sugar to alcohol. Here, it's expressed in g/cm3.

• **pH**: Also referred to as hydrogen potential, this is a numerical scale used to describe the acidity or basicity of wine. The pH of wines is largely affected by fixed acidity. As you may know, acidic solutions have a pH below 7, while basic solutions have a pH over 7. Pure water is neutral when its pH value is 7. Most wines are acidic because their pH ranges from 2.9 to 3.9.

***.* Sulfates**: Mineral salts that contain sulfur are known as slip sulfates gluten are to food, and sulfates are to wine. They are regarded as being necessary and are frequently used in decision-making. It is written in the dataset as g(potassium sulfate).

. Alcohol: Wine is an alcoholic beverage. Alcohol is formed as a result of yeast converting sugar during the fermentation process. The percentage of alcohol can vary from wine to wine. Hence it is not a surprise for this attributeto be a part of this dataset.

• **Quality:** Wine experts graded the wine quality between 0 (very bad) and 10(very excellent). The eventual quality score is the median of at least three evaluations made by the same wine experts.

• **Wine type:** Since we originally had two datasets for red and white wine, we introduced this attribute in the final merged dataset which indicates the type of wine for each data point. A wine can either be a 'red' or a 'white' wine. One of the predictive models we will build would be such that we can predict the type of wine bylooking at other wine attributes.

• **Quality label:** This is a derived attribute from the quality attribute. We bucket or group wine quality scores into three qualitative buckets namely low, medium and high.

1.2 MACHINE LEARNING

Machine Learning is extensively used in analyzing wine type and quality. Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it to learn forthemselves. The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. The primary aim is to allow computers to learn automatically without human intervention or assistance and adjust actions accordingly.

1.2.1 *SUPERVISED MACHINE LEARNING*

These algorithms can apply what has been learned in the past to new data using labelled examples to predict future events. Starting from the analysis ofa known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system is able to provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly.

1.2.2 UN*-SUPERVISED MACHINE LEARNING*

In unsupervised learning, the system looks for correlations between variables or data's hidden structure. The training data in that situation consists of examples without any associated labels. Rule of Association Machine learning came into being much more recently, and mining is more heavily influenced by database research. The process of grouping is called cluster analysis or clustering. A collection of items arranged so that members of the same (grocluster) resemble one another more closely than members of other groups do (clusters).

It is a primary function of exploratory data mining and a widely used statistical data analysis method in a variety of domains, such as bioinformatics, machine learning, pattern recognition, and image analysis.

1.2.3 REINFORCEMENT LEARNING

The phrase "Reinforcement Learning" refers to a family of learning strategies in which the system tries to pick up new information by interacting directly with its surroundings in order to optimize some kind of cumulative reward. It is crucial to note that the system lacks knowledge of how the world behaves, and the only way to learn is through trial and error (trial and error). Due to its independence from its surroundings, reinforcement learning is mostly used in autonomous systems.

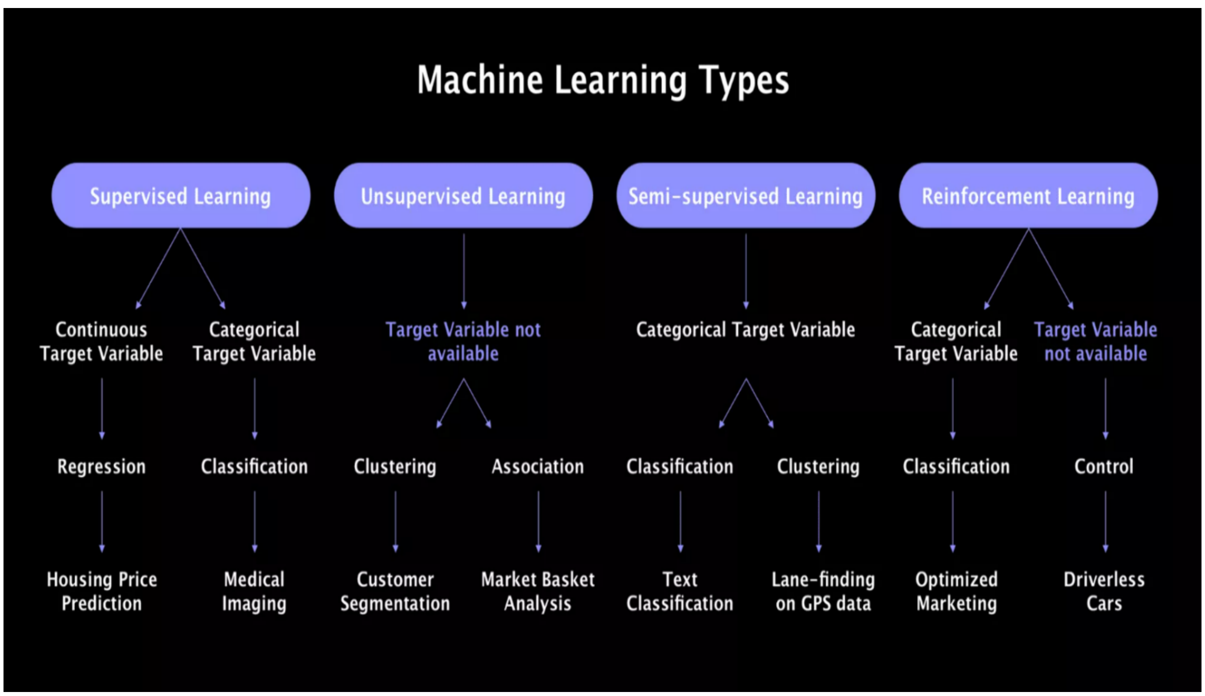


FIG 1.1: MACHINE LEARNING TYPES

CHAPTER 2

AIM AND SCOPE

2.1 SOFTWARE COMPONENTS

JUPYTER NOTE BOOK:

You may create and share documents that contain live code, mathematics, graphics, and narrative texts with the free source Jupiter Notebook program. Data processing and cleansing, numerical simulations, statistical modeling, data visualizations, machine learning, and many other uses are included.

PYTHON:

Python code is understandable by humans, which makes it easier to build models formachine learning. Since Python is a general-purpose language, it can do a set of complex machines learnimachine-learningble you to build prototypes quickly that allow you to tallows our product for machine learning purposes. And detailed history and features are explained below.

2.2 AIM OF THE PROJECT:

The primary goal of the project is to train an ML model with a given Algorithm, that can predict the Quality of wine, based on the input features given to it. The main challenge of this project is to understand the dataset, deal with missing values, use the right performance metrics for the algorithm and train the model with root mean square error for regression. Using python and python integrated modules helps to face the challenges of a dataset and make an efficient model for predicting things.

2.3 SCOPE OF THE PROJECT

This problem can be solved by a variety of machine learning (ML) techniques, but in this project, we are using the linear regression approach because it is one of the more effective ML algorithms for wine regression.

CHAPTER 3

EXPERIMENTAL OR MATERIALS AND METHODS; ALGORITHMS USED

3.1 FEATURES OF WINE QUALITY TESTING

Attibutes	Description
pH	To measure ripeness
Density	Density in gram per cm3
Alcohol	Volume of alcohol in %
Fixed Acidity	Impart sourness and resist microbial infection, measured in no.of grams of tartaric acid per dm3
Volatile Acidity	No. of grams of acetic acid per dm3 of wine
Citric Acid	No. of grams of citric acid per dm3 of wine
Residual Sugar	Remaining sugar after fermentation stops
Chlorides	No. of grams of sodium chloride per dm3 of wine
Free Sulfur dioxide	No. of grams of free sulphites per dm3 of wine
Total Sulfur dioxide	No. of grams of total sulphite
Sulphates	No. of grams of potassium sulphate per dm3 of wine
Quality	Target variable, 1-10 value

The UCI machine learning repository, which has a sizable collection of datasets that have been utilised by the machine learning community, is where the red wine and white wine datasets that were used in this research were obtained. Two excel files relevant to red wine and white wine variations of the Portuguese "Vinho Verde" wine are included in the dataset (Cortez et al., 2009). The white wine dataset has 4898 cases, whereas the red wine dataset has 1599. Both datasets contain 1 output variable (based on sensory data) called quality and 11 input variables (based on physicochemical tests):

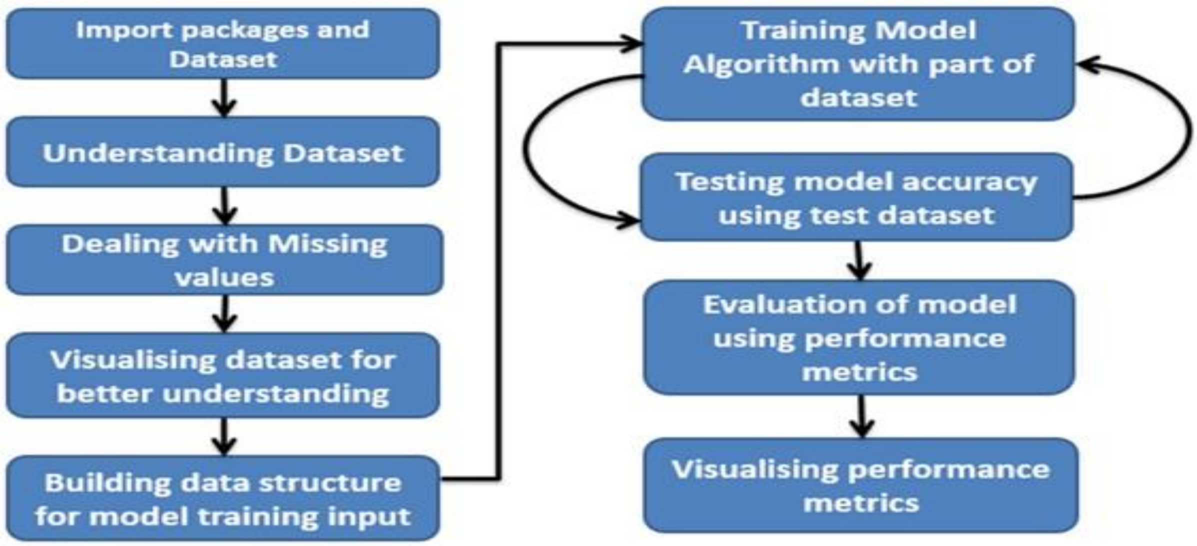


***FIG:* 3.1: Architecture map of the project**

Dataset Description:

The two datasets are connected to Portuguese red wine. Consult [Web Link] or the source [Cortez et al., 2009] for more information. . Only physicochemical (input) and sensory (output) variables are available due to logistical and privacy concerns (e.g., there is no data about grape types, wine brand, wine selling price, etc.). These datasets can be used to perform regression or classification tasks. The classes are not balanced and are in order The few exceptional or subpar wines could be identified using outlier detection techniques.

1) volatile acidity

2) citric acid

3) residual sugar

4) chlorides

6) free sulfur dioxide

7) total sulfur dioxide

8) density

9) pH

10) alcohol Output variable (based on sensory data):

Understanding dataset via tables

These features are used in the dataset are understood by definition but we also needto understand the structure of the dataset, how data is represented in the tabular form, find out the missing values, and fill the missing values. This data is representedin the below tables.

Table 3.1: Sample records of wine quality at the beginning of the dataset

	type	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	white	7.0	0.27	0.36	20.7	0.045	45.0	170.0	1.0010	3.00	0.45	8.8	6
1	white	6.3	0.30	0.34	1.6	0.049	14.0	132.0	0.9940	3.30	0.49	9.5	6
2	white	8.1	0.28	0.40	6.9	0.050	30.0	97.0	0.9951	3.26	0.44	10.1	6
3	white	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9.9	6
4	white	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9.9	6

Table 3.2: Sample records of wine quality in the ending of the dataset

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
6492	6.2	0.600	0.08	2.0	0.090	32.0	44.0	0.99490	3.45	0.58	10.5	5
6493	5.9	0.550	0.10	2.2	0.062	39.0	51.0	0.99512	3.52	NaN	11.2	6
6494	6.3	0.510	0.13	2.3	0.076	29.0	40.0	0.99574	3.42	0.75	11.0	6
6495	5.9	0.645	0.12	2.0	0.075	32.0	44.0	0.99547	3.57	0.71	10.2	5
6496	6.0	0.310	0.47	3.6	0.067	18.0	42.0	0.99549	3.39	0.66	11.0	6

Table: -3.3 estimating Null values

```
# Number of missing values in each column of training data
missing_val_count_by_column = (df.isnull().sum())
print(missing_val_count_by_column[missing_val_count_by_column > 0])
```

```
fixed acidity       10
volatile acidity     8
citric acid          3
residual sugar       2
chlorides            2
pH                   9
sulphates            4
dtype: int64
```

Table: -3. 4 Describing about datatype

```
: type                        object
  fixed acidity              float64
  volatile acidity           float64
  citric acid                float64
  residual sugar             float64
  chlorides                  float64
  free sulfur dioxide        float64
  total sulfur dioxide       float64
  density                    float64
  pH                         float64
  sulphates                  float64
  alcohol                    float64
  quality                      int64
  dtype: object
```

Table 3.5 Description of the Dataset

The number of records, minimum value, maximum value, standard deviation, mean, 25% of max value, 50%(median) of the max value, 75% of the max values on the min-max range values of the dataset are represented in Table 3.5.

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	c
count	6487.000000	6489.000000	6494.000000	6495.000000	6495.000000	6497.000000	6497.000000	6497.000000	6488.000000	6493.000000	6497.000000	6497.0
mean	7.216579	0.339691	0.318722	5.444326	0.056042	30.525319	115.744574	0.994697	3.218395	0.531215	10.491801	5.8
std	1.296750	0.164649	0.145265	4.758125	0.035036	17.749400	56.521855	0.002999	0.160748	0.148814	1.192712	0.8
min	3.800000	0.080000	0.000000	0.600000	0.009000	1.000000	6.000000	0.987110	2.720000	0.220000	8.000000	3.0
25%	6.400000	0.230000	0.250000	1.800000	0.038000	17.000000	77.000000	0.992340	3.110000	0.430000	9.500000	5.0
50%	7.000000	0.290000	0.310000	3.000000	0.047000	29.000000	118.000000	0.994890	3.210000	0.510000	10.300000	6.0
75%	7.700000	0.400000	0.390000	8.100000	0.065000	41.000000	156.000000	0.996990	3.320000	0.600000	11.300000	6.0
max	15.900000	1.580000	1.660000	65.800000	0.611000	289.000000	440.000000	1.038980	4.010000	2.000000	14.900000	9.0

3.2 CORRELATION(Graphs)

Correlation is a statistical term portraying how much two variables move in coordination with each other. It can also be said as how two variables are dependent on each other. If the two variables move in a similar way, those variables are said to have a positive correlation. If they move in inverse ways,they have a negative correlation. The correlation among the features of the rice in the dataset has been shown in figure 3.2

The wine-type labels in the features are called class. It is shown in the graph for the number of wine types and their count in figure 3.4.

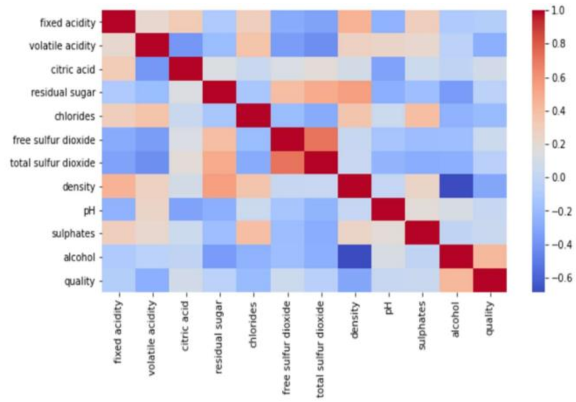


Fig:3.2:Correlationamongfeaturesof wine quality

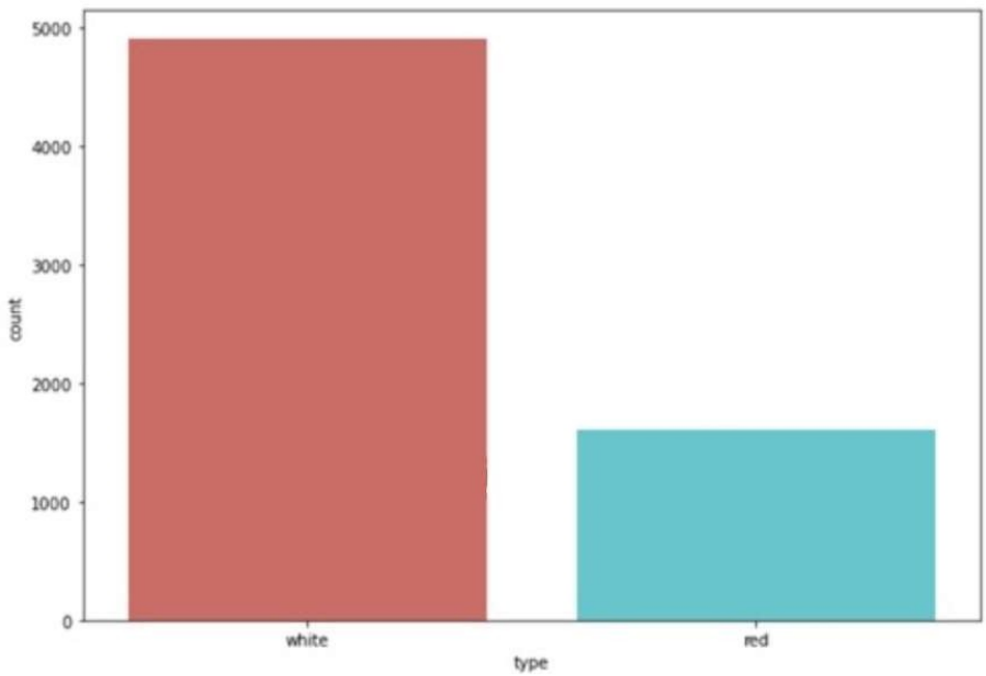


Fig:3.3: graph of to be predicted class

If I plot feature vs feature graph for all features in the dataset of two wine types there will be more similarities in the structure of graphs but the difference in coordinates. It is due to the main physical structure of wine being the same but in terms of directions, they are different. This significant difference in range b/w two wine quality types helps to predict their classification with Random Forest Algorithm.

3.2 LINEAR REGRESSION

Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as sales, salary, age, product price, quality etc.

Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called as linear regression. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.

The linear regression model provides a sloped straight line representing the relationship between the variables. Consider the below image:

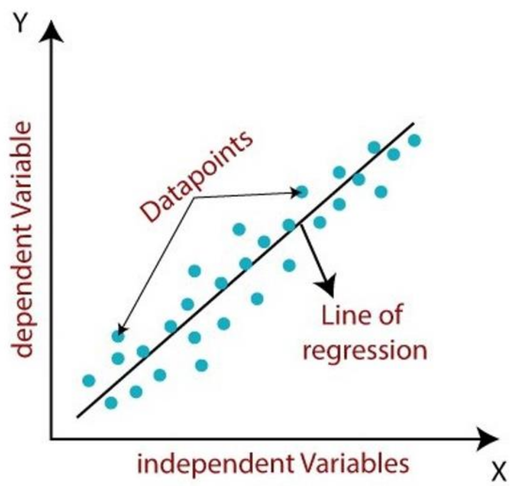


Fig 3.4 LINEAR REGRESSION GRAPH

Mathematically, we can represent a linear regression as: y= a0+a1x+ ε

Here,

Y = Dependent Variable
X = Independent Variable

a0 = intercept of the line
a1 = Linear regression
ε = random error

The values for x and y variables are training datasets for Linear Regression model representation.

3.3 DATA PROCESSING METHODS

For making automated decisions on model selection i need to quantify the performance of our model and give it a score. For that reason, for the classifiers, we are using F1 score which combines two metrics: Precision which expresses how accurate the model was on predicting a certain class and Recall which expresses the inverse of the regret of missing out instances which are misclassified. Since we have multiple classes, we have multiple F1 scores. We will be using the unweighted mean of the F1 scores for our final scoring. This is a business decision because we want ourmodels to get optimized to classify instances that belong to the minority side, such aswine quality of 3 or 8 equally well with the rest of the qualities that are represented ina larger number. For the regression task we are scoring based on the coefficient of determination, which is basically a measurement of whether the predictions and theactual values are highly correlated. The larger this coefficient the better. Forregressors we can also get F1 score if we first round our prediction.

3.3.1 Splitting for Testing:

I am keeping 20% of our dataset to treat it as unseen data and be ableand test the performance of our models. I am splitting our dataset in a way such that all of the wine qualities are represented proportionally equally in both training and testing dataset. Other than that the selection is being done randomly with uniform distribution. Various classification and regression algorithmsare used to fit the model. The algorithms used in this paper are as follows:

Splitting for Testing:

I am keeping 20% of our dataset to treat it as unseen data and be ableand test the performance of our models. I am splitting our dataset in a way such that all of the wine qualities are represented proportionally equally in both training and testing dataset.

For classification:

1. Random Forest
2. Decision Trees classifier
3. Support Vector Machine classifier
4. Stochastic gradient descent
5. Logistic Regression classifier

3.4.2. Preprocessing:

Label Encoding is used to convert the labels into numeric form so as to convertit into the machine-readable form. It is an important pre-processing step for the structured dataset in supervised learning. We have used label encoding to label the quality of data as good or bad. Assigning 6-10 to good and 0-5 to bad.

3.4.3 Bagging method:

Bagging, also known as Bootstrap aggregating, is an ensemble learning technique that helps to improve the performance and accuracy of machine learning algorithms. It is used to deal with bias-variance trade-offs and reduces the variance of a prediction model. Bagging avoids overfitting of data and is used for both regression and classification models, specifically for decision tree algorithms.

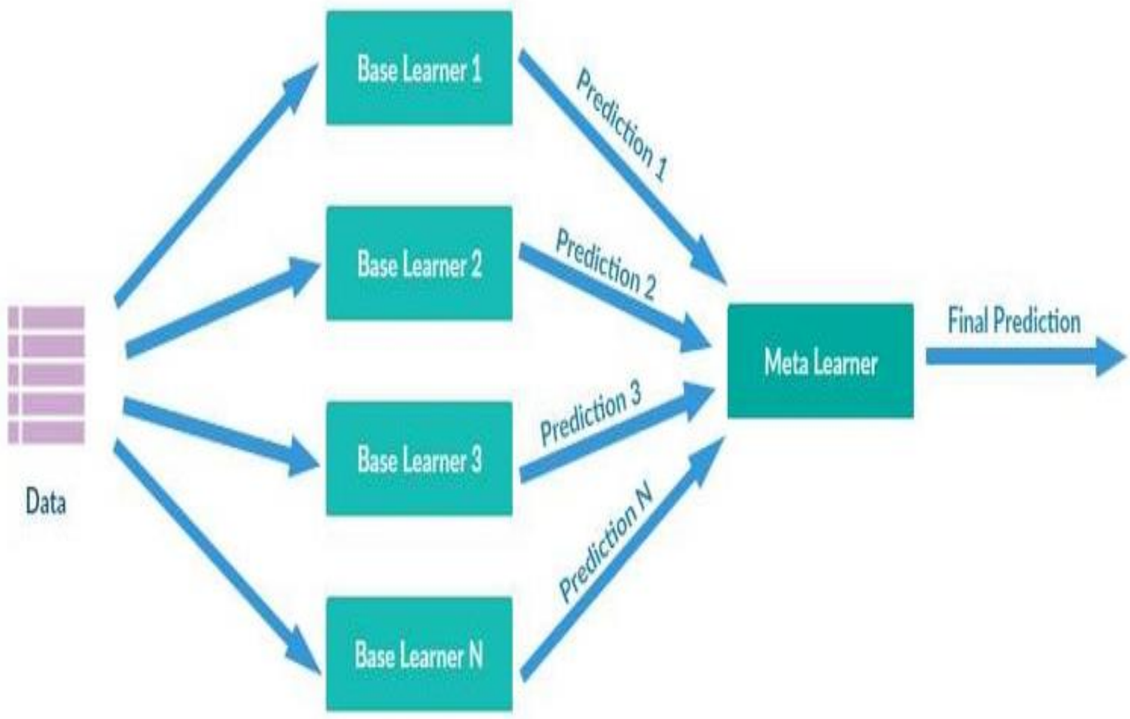


Fig-3.5: - The method in Machine Learning: Bagging

Steps to Perform Bagging:

- Consider there are n observations and m features in the training set. You need to select a random sample from the training dataset without replacement
- A subset of m features is chosen randomly to create a model using sample observations
- The feature offering the best split out of the lot is used to split the nodes.
- The tree is grown, so you have the best root nodes.
- The above steps are repeated n times. It aggregates the output of individualdecision trees to give the best prediction.

Advantages of Bagging in Machine Learning:

- Bagging minimizes the overfitting of data.
- It improves the model's accuracy.
- It deals with higher dimensional data efficiently.

3.1 IMPORT THE REQUIRED LIBRARIES:

3.5 IMPORTING LIBRARIES

Importing the Relevant Libraries

In [1]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
import seaborn as sns
from sklearn import metrics
sns.set()
```

Pandas: means it is an open-source Python package that is most widely used for data science/data analysis and machine learning tasks. Pandas is very fast.

NumPy: 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 and much more.

Seaborn:

Seaborn is an open-source Python library built on top of matplotlib. It is used for datavisualization and exploratory data analysis. Seaborn works easily with data frames and the Pandas library. The graphs created can also be customized easily.

Matplotlib: Matplotlib is one of the most popular Python packages used fordata visualization. It is a cross-platform library for making 2D plots from data in arrays.

Sklearn: Scikit-learn (Sklearn) is the most useful and robust library formachine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression etc.

- Next step in the coding part to call the dataset by using pandas by the usingthe below.
- As you can see the above the data set is read by (data=pd.read_csv("covtype.csv")) by read function Call the data set.
- CSV stands for "Comma-Separated Values" files; it is a file format which allows us to save the tabular data, such as spreadsheets. It is useful for hugedatasets and can use these datasets in programs

3.5.1 LOAD THE DATA

Data. shape (): In the line the code shows the dimensions of the dataset. In simple words it tells how many rows and columns are there in your dataset

Reading and Understanding the Data

```
In [3]: df = pd.read_csv('winequalityN.csv')
        df.head()
```

Out[3]:

	type	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	white	7.0	0.27	0.36	20.7	0.045	45.0	170.0	1.0010	3.00	0.45	8.8	6
1	white	6.3	0.30	0.34	1.6	0.049	14.0	132.0	0.9940	3.30	0.49	9.5	6
2	white	8.1	0.28	0.40	6.9	0.050	30.0	97.0	0.9951	3.26	0.44	10.1	6
3	white	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9.9	6
4	white	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9.9	6

```
In [4]: df.shape
Out[4]: (6497, 13)
```

3.5.2 TRAIN_TEST_SPLIT

The train-test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to trainthe model.

It is a fast and easy procedure to perform, the results of which allow you to compare the performance of machine learning algorithms for your predictive modeling problem. Although simple to use and interpret, there are times when theprocedure should not be used, such as when you have a small dataset and situations where additional configuration is required, such as when it is used for classification and the dataset is not balanced.

- Train: 70% and Test: 30%
- Train: 80% and Test: 20%
- Train: 90% and Test: 10%

Here we used Train: 80% and Test: 20.

- Training data is used for learning the parameters of the model.
- Validation data is not used of learning but is used for deciding what type of model and what amount of regularization works best. —
- Test data is used to get a final, unbiased estimate of how well the network works. We expect this estimate to be worse than on the validation data.

```
In [30]: from sklearn.model_selection import train_test_split
         x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=22)
```

USING LINEAR REGRESSION

```
In [68]: Lin=LinearRegression()
         Lin.fit(x_train,y_train)

Out[68]: LinearRegression()
```

CHAPTER 4

RESULTS AND DISCUSSION, PERFORMANCEANALYSIS

4.1 ANALYSIS OF WINE QUALITY DATA

Image of red wine In the second example of data mining for knowledge discovery, we consider a set of observations on a number of red and white wine varieties involving their chemical properties and ranking by tasters. Wine industryshows 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. Pricing of wine depends on such a volatile factor to some extent. 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.

Two datasets are available of which one dataset is on red wine and have 1599 different varieties and the other is on white wine and have 4898 varieties. Only white wine data is analyzed. All wines are produced in a particular area of Portugal. 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 thewines 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 tasters.

4.2 ROOT MEAN SQUARE ERROR(RMSE): -

Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is aroundthe line of best fit. Root means square error is commonly used in climatology, forecasting, and regression analysis to verify experimental results.

- We are able to achieve RMSE of around 0.75 on the white wine testing data, and upon reusing the same model on the red wine test data, we achieve RMSE of around 0.77, which is really good considering feature reuse. This means that our model has done a good job on generalizing the white wine dataset. We have also observed that while performing gradient descent on the red wine training data we are able to achieve RMSE of around 0.65. and using that model on the white wine testing datayields a RMSE of 0.81, which means training the data on the red wine training data very slightly overfits the red wine data, whereas the model obtained by training on the white wine training data generalized the entire wine dataset better. Furthermore, on training a classifier to distinguish red and white wines, we are able to achieve around 90 percent accuracy on the test data. This implies that the datasets of white and red wines have some uniqueness in their features that distinguish them to their own groups. So, if the white wine model were too overfit the white wine data, you would have a really poor performance (an RMSE of greater than 1) onthe red wine testing data. But we find that we are achieving an almost similar performance on both the testing data. This implies that the model is really well generalized.

CHAPTER 5

SUMMARY AND CONCLUSIONS

This study's specific goal is to examine how physical and chemical characteristics, such as alcohol content, chloride levels, sulphate concentrations, etc., affect wine quality. This study uses a variety of physicochemical factors to analyses the different types and qualities of wine. Using samples of red and white wine, two datasets were produced. The statistically significant attribute that affects wine quality out of the thirteen attributes is a crucial result. the model that emphasizes the important factor in both sets. This finding is useful for predicting quality and productivity by looking at those characteristics. Utilizing three machine learning algorithms—decision tree, random forest, and extreme gradient boosting—analyze the wine type using logistic regression and the quality. Compared to earlier methods, the results are more precise.

- After having obtained all the results through our models and plots, these aresome things we can say about this problem and solution:
- The vast majority of wines get a quality rating of five or six, while having goodand bad wines seems more unlikely. There seem not to be any excellent wines (>8) on this database.
- From the very first moment we saw there weren't strong correlations between features and quality, that's why it's hard to make an accurate prediction using regression algorithms. That said, alcohol, sulphates, citric acid features are the ones that correlate the most positively while volatile acidity is the one correlating the most negatively.
- Applying the concept 1-off Accuracy gives us much better results.
- Random Forest and Linear Regression seem to be the best fitting modelswhen solving this problem using regression.

- Therefore, in the classification algorithms by selecting the appropriate features and balancing the data can improve the performance of the model. The interest has been increased in wine industry in recent years which demands growth in this industry. Therefore, companies are investing in new technologies to improve wine production and selling. In this direction, wine quality certification plays a very important role for both processes and it requires wine testing by human experts. This paper explores the usage of machine learning techniques in two ways. Firstly, how linear regression determines important features for prediction. Secondly, the usage of neural network and support vector machine in predicting the values. The benchmark Wine dataset is used for all experiments. This dataset has two parts: Red Wine and White Wine data. Red wine contains 1599 samples and white wine contains 4898 samples. Both red and white wine dataset consists of 12 physicochemical characteristics. One (quality) is dependent variable and other 11 are predictors. The experiments shows that the value of dependent variable can be predicted more accurately if only important features are considered in prediction rather than considering all features. In future, large dataset can be taken for experiments and other machine learning techniques may be explored for wine quality prediction.

REFERENCES:

1. Pandas.dataframe. pandas. data Frame - pandas 1.3.4 documentation. (n.d.). Retrieved November 2, 2021, from https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.html.

2. Sklearn.ensemble. Linear Regression. scikit. (n.d.). Retrieved November 2, 2021, from https://scikitlearn.org/stable/modules/generated/sklearn.linear_model.LinearRegresson.html

3. Sklearn.metrics.roc_auc_score. scikit. (n.d.). RetrievedNovember2,2021, fromhttps://scikitlearn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html

APPENDIX

PROJECT CODE:

Importing the Relevant Libraries

In [86]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [34]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
import seaborn as sns
from sklearn import metrics
sns.set()
```

Reading and Understanding the Data

In [35]:
```
df = pd.read_csv('winequalityN.csv')
df.head()
```

Out[35]:

	type	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	white	7.0	0.27	0.36	20.7	0.045	45.0	170.0	1.0010	3.00	0.45	8.8	6
1	white	6.3	0.30	0.34	1.6	0.049	14.0	132.0	0.9940	3.30	0.49	9.5	6
2	white	8.1	0.28	0.40	6.9	0.050	30.0	97.0	0.9951	3.26	0.44	10.1	6
3	white	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9.9	6
4	white	7.2	0.23	0.32	8.5	0.058	47.0	186.0	0.9956	3.19	0.40	9.9	6

In [36]:
```
df.shape
```

Out[36]: (6497, 13)

Basic Data study

```
In [37]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6497 entries, 0 to 6496
Data columns (total 13 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   type                  6497 non-null   object
 1   fixed acidity         6487 non-null   float64
 2   volatile acidity      6489 non-null   float64
 3   citric acid           6494 non-null   float64
 4   residual sugar        6495 non-null   float64
 5   chlorides             6495 non-null   float64
 6   free sulfur dioxide   6497 non-null   float64
 7   total sulfur dioxide  6497 non-null   float64
 8   density               6497 non-null   float64
 9   pH                    6488 non-null   float64
 10  sulphates             6493 non-null   float64
 11  alcohol               6497 non-null   float64
 12  quality               6497 non-null   int64
dtypes: float64(11), int64(1), object(1)
memory usage: 660.0+ KB
```

```
In [38]: df.isnull().sum()

Out[38]: type                     0
         fixed acidity           10
         volatile acidity         8
         citric acid              3
         residual sugar           2
         chlorides                2
         free sulfur dioxide      0
         total sulfur dioxide     0
         density                  0
         pH                       9
         sulphates                4
         alcohol                  0
         quality                  0
         dtype: int64
```

In [39]:
```
df=df.fillna(df.mean())
```

In [40]:
```
df
```

Out[40]:

	type	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	white	7.0	0.270	0.36	20.7	0.045	45.0	170.0	1.00100	3.00	0.450000	8.8	6
1	white	6.3	0.300	0.34	1.6	0.049	14.0	132.0	0.99400	3.30	0.490000	9.5	6
2	white	8.1	0.280	0.40	6.9	0.050	30.0	97.0	0.99510	3.26	0.440000	10.1	6
3	white	7.2	0.230	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.400000	9.9	6
4	white	7.2	0.230	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.400000	9.9	6
...	...	...	...	...	...	...	...	...	...	...	...	...	...
6492	red	6.2	0.600	0.08	2.0	0.090	32.0	44.0	0.99490	3.45	0.580000	10.5	5
6493	red	5.9	0.550	0.10	2.2	0.062	39.0	51.0	0.99512	3.52	0.531215	11.2	6
6494	red	6.3	0.510	0.13	2.3	0.076	29.0	40.0	0.99574	3.42	0.750000	11.0	6
6495	red	5.9	0.645	0.12	2.0	0.075	32.0	44.0	0.99547	3.57	0.710000	10.2	5
6496	red	6.0	0.310	0.47	3.6	0.067	18.0	42.0	0.99549	3.39	0.660000	11.0	6

6497 rows × 13 columns

In [41]:
```
df.isnull().sum()
```

Out[41]:
```
type                    0
fixed acidity           0
volatile acidity        0
citric acid             0
residual sugar          0
chlorides               0
free sulfur dioxide     0
total sulfur dioxide    0
density                 0
pH                      0
sulphates               0
alcohol                 0
quality                 0
dtype: int64
```

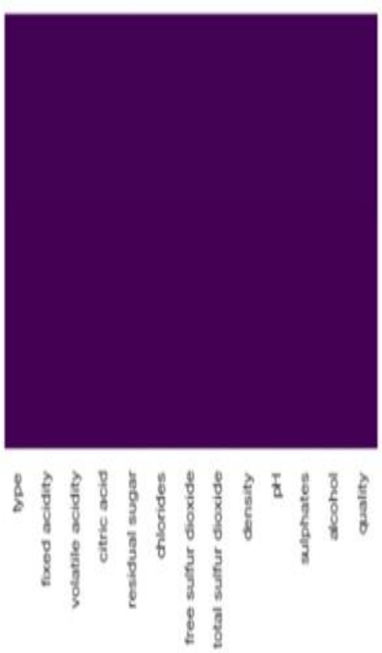
In [42]:
```
df.describe()
```

Out[42]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	q
count	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.000000	6497.0
mean	7.216579	0.339691	0.318722	5.444326	0.056042	30.525319	115.744574	0.994697	3.218395	0.531215	10.491801	5.8
std	1.295751	0.164548	0.145231	4.757392	0.035031	17.749400	56.521855	0.002999	0.160637	0.148768	1.192712	0.8
min	3.800000	0.080000	0.000000	0.600000	0.009000	1.000000	6.000000	0.987110	2.720000	0.220000	8.000000	3.0
25%	6.400000	0.230000	0.250000	1.800000	0.038000	17.000000	77.000000	0.992340	3.110000	0.430000	9.500000	5.0
50%	7.000000	0.290000	0.310000	3.000000	0.047000	29.000000	118.000000	0.994890	3.210000	0.510000	10.300000	6.0
75%	7.700000	0.400000	0.390000	8.100000	0.065000	41.000000	156.000000	0.996990	3.320000	0.600000	11.300000	6.0
max	15.900000	1.580000	1.660000	65.800000	0.611000	289.000000	440.000000	1.038980	4.010000	2.000000	14.900000	9.0

In [43]:
```
sns.heatmap(df.isnull(),yticklabels=False , cbar=False,cmap='viridis')
```

Out[43]: <AxesSubplot:>



Examine Duplicate values

In [44]:
```
duplicate = df.duplicated()
print(duplicate.sum())
df[duplicate]
```

1168

Out[44]:

	type	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
4	white	7.2	0.230	0.32	8.50	0.058	47.0	186.0	0.99560	3.19	0.40	9.9	6
5	white	8.1	0.280	0.40	6.90	0.050	30.0	97.0	0.99510	3.26	0.44	10.1	6
7	white	7.0	0.270	0.36	20.70	0.045	45.0	170.0	1.00100	3.00	0.45	8.8	6
8	white	6.3	0.300	0.34	1.60	0.049	14.0	132.0	0.99400	3.30	0.49	9.5	6
39	white	7.3	0.240	0.39	17.95	0.057	45.0	149.0	0.99990	3.21	0.36	8.6	5
...	...	...	...	...	...	...	...	...	...	...	...	...	...
6461	red	7.2	0.695	0.13	2.00	0.076	12.0	20.0	0.99546	3.29	0.54	10.1	5
6462	red	7.2	0.695	0.13	2.00	0.076	12.0	20.0	0.99546	3.29	0.54	10.1	5
6465	red	7.2	0.695	0.13	2.00	0.076	12.0	20.0	0.99546	3.29	0.54	10.1	5
6479	red	6.2	0.560	0.09	1.70	0.053	24.0	32.0	0.99402	3.54	0.60	11.3	5
6494	red	6.3	0.510	0.13	2.30	0.076	29.0	40.0	0.99574	3.42	0.75	11.0	6

1168 rows × 13 columns

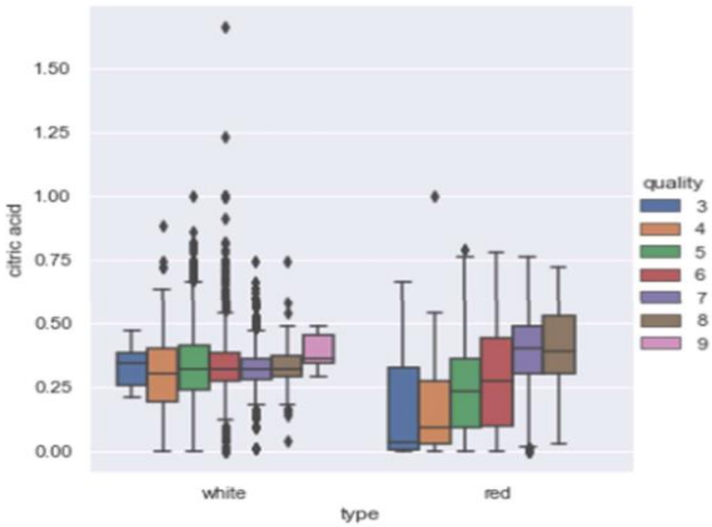
In [45]:
```
df.drop_duplicates(inplace=True)
```

In [46]:
```
df.shape
```

Out[46]: (5329, 13)

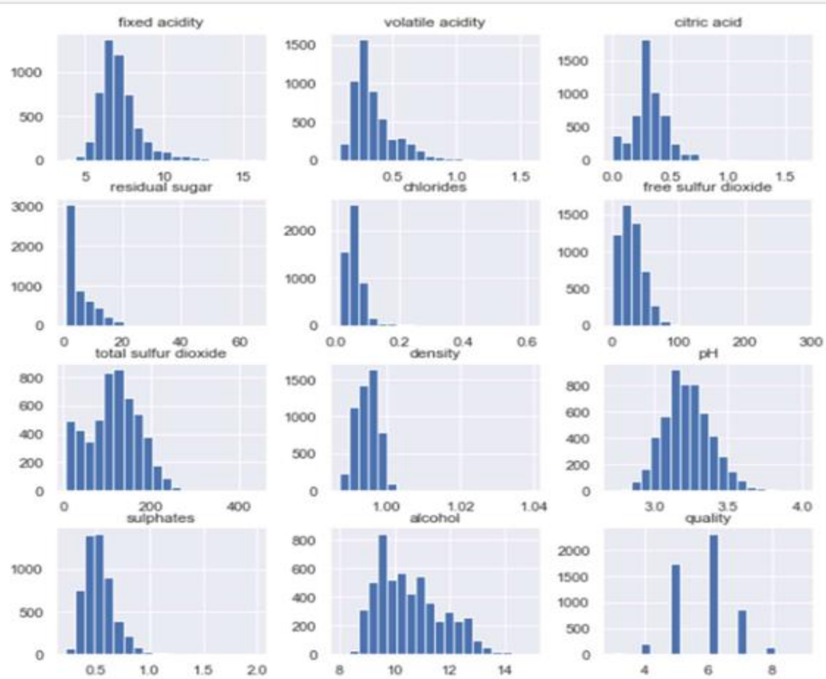
Outliers and Visualization

```
In [47]: sns.catplot(x="type", y="citric acid", kind="box", hue="quality", data=df)
Out[47]: <seaborn.axisgrid.FacetGrid at 0x2457e2740d0>
```



HISTOGRAM

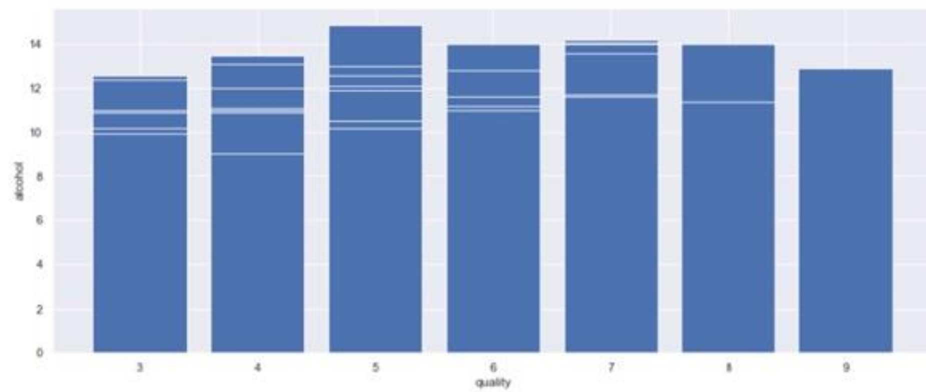
```
In [79]: df.hist(bins=20,figsize=(10,10))
         plt.show()
```



BAR GRAPH BETWEEN ALCOHOL% AND QUALITY

BAR GRAPH BETWEEN ALCOHOL% AND QUALITY

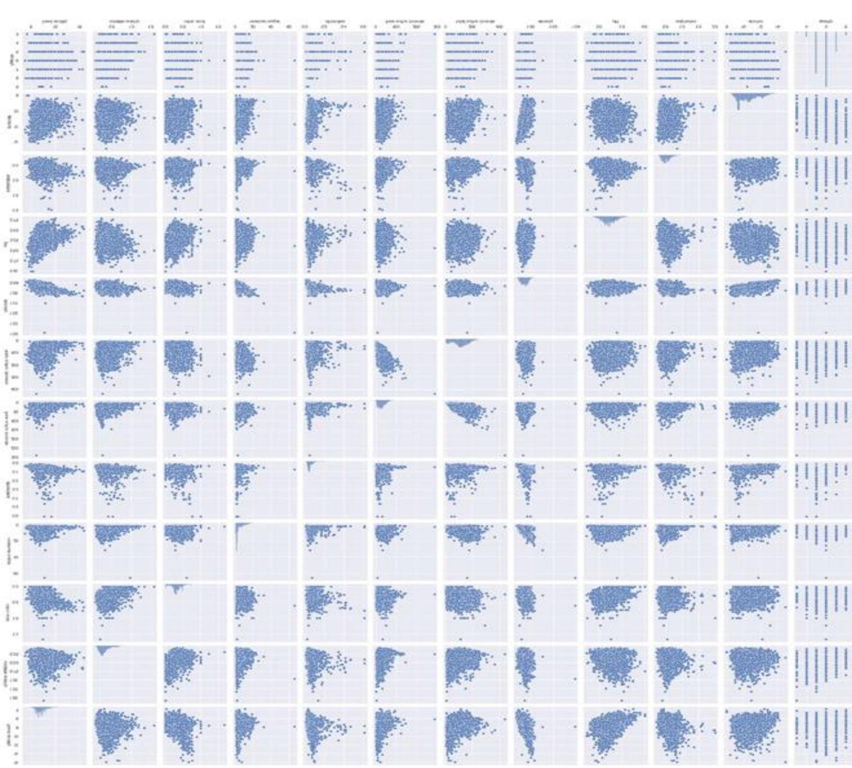
```
In [81]: plt.figure(figsize=[15,6])
         plt.bar(df['quality'],df['alcohol'])
         plt.xlabel('quality')
         plt.ylabel('alcohol')
         plt.show()
         #GRAPH BETWEEN ALCOHOL% AND QUALITY OF WINE
```



PAIRPLOT

```
In [78]: sns.pairplot(df)

Out[78]: <seaborn.axisgrid.PairGrid at 0x245764cd6d0>
```



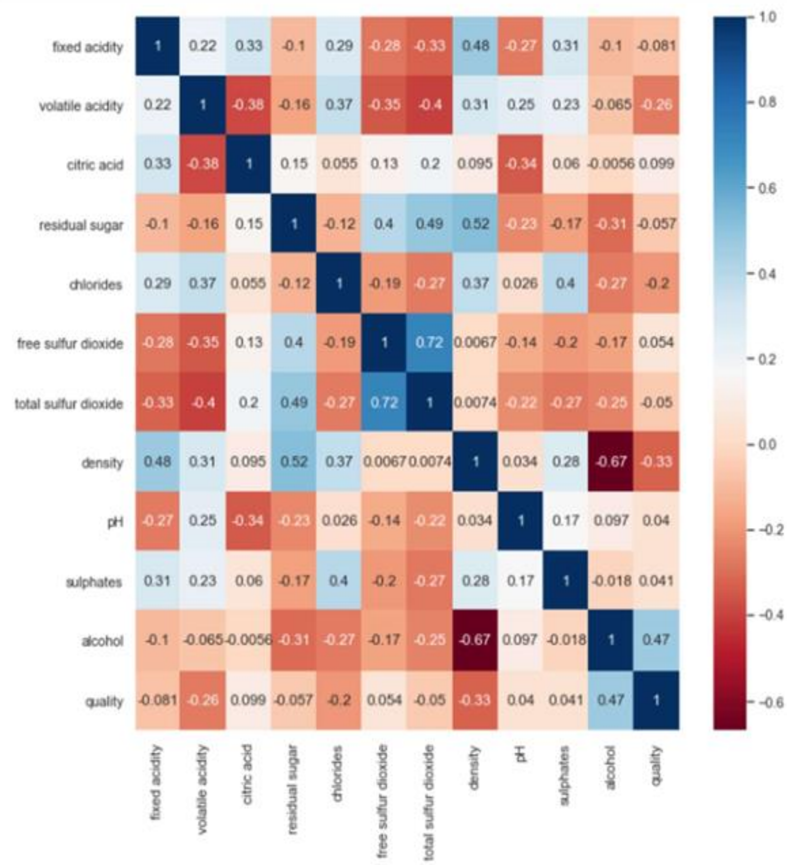
Correlation Matrix

In [53]:
```
correlations = df_out.corr()['quality'].drop('quality')
print(correlations)
```

```
fixed acidity           -0.095044
volatile acidity        -0.230909
citric acid              0.102786
residual sugar          -0.056404
chlorides               -0.261122
free sulfur dioxide      0.075533
total sulfur dioxide    -0.068078
density                 -0.333474
pH                       0.051989
sulphates                0.058371
alcohol                  0.453596
Name: quality, dtype: float64
```

HEAT MAP

In [54]:
```
plt.figure(figsize=(10,10))
sns.heatmap(df.corr(),annot=True,cmap='RdBu')
plt.show()
```



```
In [55]: correlations.sort_values(ascending=False)
```

```
Out[55]: alcohol                 0.453596
         citric acid             0.102786
         free sulfur dioxide     0.075533
         sulphates               0.058371
         pH                      0.051989
         residual sugar         -0.056404
         total sulfur dioxide   -0.068078
         fixed acidity          -0.095044
         volatile acidity       -0.230909
         chlorides              -0.261122
         density                -0.333474
         Name: quality, dtype: float64
```

```
In [56]: abs_corrs = correlations.abs()
         print(abs_corrs)
```

```
fixed acidity           0.095044
volatile acidity        0.230909
citric acid             0.102786
residual sugar          0.056404
chlorides               0.261122
free sulfur dioxide     0.075533
total sulfur dioxide    0.068078
density                 0.333474
pH                      0.051989
sulphates               0.058371
alcohol                 0.453596
Name: quality, dtype: float64
```

```
In [57]: def get_features(correlation_threshold):
             abs_corrs = correlations.abs()
             high_correlations = abs_corrs[abs_corrs > correlation_threshold].index.values.tolist()
             return high_correlations
```

```
In [58]: #Data preprocessing
         features = get_features(0.05)
         print(features)
         x = df_out[features]
```

```
['fixed acidity', 'volatile acidity', 'citric acid', 'residual sugar', 'chlorides', 'free sulfur dioxid
e', 'density', 'pH', 'sulphates', 'alcohol']
```

```
In [59]: x
```

Out[59]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol
1	6.3	0.300	0.34	1.6	0.049	14.0	132.0	0.99400	3.30	0.490000	9.5
2	8.1	0.280	0.40	6.9	0.050	30.0	97.0	0.99510	3.26	0.440000	10.1
3	7.2	0.230	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.400000	9.9
6	6.2	0.320	0.16	7.0	0.045	30.0	136.0	0.99490	3.18	0.470000	9.6
9	8.1	0.220	0.43	1.5	0.044	28.0	129.0	0.99380	3.22	0.450000	11.0
...	...	...	...	...	...	...	...	...	...	...	...
6491	6.8	0.620	0.08	1.9	0.068	28.0	38.0	0.99651	3.42	0.820000	9.5
6492	6.2	0.600	0.08	2.0	0.090	32.0	44.0	0.99490	3.45	0.580000	10.5
6493	5.9	0.550	0.10	2.2	0.062	39.0	51.0	0.99512	3.52	0.531215	11.2
6495	5.9	0.645	0.12	2.0	0.075	32.0	44.0	0.99547	3.57	0.710000	10.2
6496	6.0	0.310	0.47	3.6	0.067	18.0	42.0	0.99549	3.39	0.660000	11.0

4088 rows × 11 columns

```
In [61]: y = df_out['quality']

In [62]: y

Out[62]: 1       6
         2       6
         3       6
         6       6
         9       6
                ..
         6491    6
         6492    5
         6493    6
         6495    5
         6496    6
         Name: quality, Length: 4088, dtype: int64

In [84]: # TRAIN TEST SPLIT
         from sklearn.model_selection import train_test_split
         x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=22)

In [85]: y_test.shape
         x_test.shape

Out[85]: (1227, 11)
```

LINEAR REGRESSION

```
In [68]: Lin=LinearRegression()
         Lin.fit(x_train,y_train)

Out[68]: LinearRegression()

In [69]: Lin.intercept_

Out[69]: 63.072123228341056

In [70]: Lin.coef_

Out[70]: array([ 6.50613262e-02, -1.06678731e+00,  9.18495158e-02,  3.87955896e-02,
                -7.94242480e-01,  8.58851720e-03, -2.38090450e-03, -6.32554045e+01,
                 7.64217691e-01,  8.86143563e-01,  2.26556607e-01])

In [71]: pred_y = Lin.predict(x_test)

In [72]: pred_y

Out[72]: array([5.62016967, 5.38866076, 5.88377747, ..., 5.48144545, 5.69909233,
                5.27357569])

In [73]: test_rmse = metrics.mean_squared_error(pred_y, y_test) ** 0.5
         test_rmse

Out[73]: 0.6420910486182664

In [74]: predicted_data = np.round_(pred_y)
         predicted_data

Out[74]: array([6., 5., 6., ..., 5., 6., 5.])
```

In [75]:
```
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, pred_y))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, pred_y))
rmse = np.sqrt(metrics.mean_squared_error(y_test, pred_y))
print('Root Mean Squared Error:',rmse)
```

```
Mean Absolute Error: 0.5095722547565351
Mean Squared Error: 0.41228091471570494
Root Mean Squared Error: 0.6420910486182664
```

In [76]:
```
import math
```

In [77]:
```
print('Correlation: ', math.sqrt(Lin.score(x_train,y_train)))
```

```
Correlation:  0.5401719160406432
```