COURSE CODE : INT-254

PROJECT REPORT

ON

Predicting Wine Quality using Wine Quality Dataset

Submitted in partial fulfilment of the requirements for the assignment

Of

B. Tech (INT-254)

In

Computer Science & Engineering (Hons.)

Submitted to:

Submitted by:

1. Nagam Haritha (12004670)
2. Bijja Thimothy Lipsika (12016706)

Under the Guidance of

Dr.Dhanpratap Singh

STUDENT DECLARATION

This is to declare that this report has been written by me/us. No part of the report is copied from other sources. All information included from other sources have been duly acknowledged. I/We aver that if any part of the report is found to be copied, I/we are shall take full responsibility for it.

SIGNATURE OF STUDENT:

Haritha

Lipsika

TABLE OF CONTENTS

BONAFIDE CERTIFICATE

Certified that this project report "**Predicting Wine Quality using Wine Quality Dataset**"is the bonafide work of "**Haritha and Lipsika"** who carried out the project work under my supervision.

Signature of the Supervisor:

Dr.Dhanpratap Singh

(25706)

INTRODUCTION

Here we will predict the quality of wine on the basis of given features. We use the wine quality dataset available on Internet for free. This dataset has the fundamental features which are responsible for affecting the quality of the wine. By the use of several Machine learning models, we will predict the quality of the wine.

if you think deep down then you just notice that we are discussing wine, above quote seems to be right because all over the world wine was soo popular among people, and 5% of the population doesn't know what is wine? sounds good.

We definitely came across the fruit **graphs**, which is soo sweet on the test but graphs are not just to eat, they are used to make different types of things. Wine is one of them **Wine is an alcoholic drink that is made up of fermented grapes**. If you have come across wine then you will notice that wine has also their type they are red and white wine this was because of different varieties of graphs.

Importing libraries and Dataset:

Pandas is a useful library in data handling.

Numpy library used for working with arrays.

Seaborn/Matplotlib are used for data visualisation purpose.

Sklearn – This module contains multiple libraries having pre-implemented functions to perform tasks from data preprocessing to model development and evaluation.

XGBoost – This contains the eXtreme Gradient Boosting machine learning algorithm which is one of the algorithms which helps us to achieve high accuracy on predictions.

```
# import pandas
import pandas as pd

# import numpy
import numpy as np

# import seaborn
import seaborn as sb

# import matplotlib
import matplotlib.pyplot as plt
```

Let's we take brief about these libraries, **pandas** are used for data analysis **NumPy** is for n-dimensional array **seaborn** and **matplotlib** both have similar functionalities which are used for visualization.

DESCRIPTION OF DATASET

If you download the dataset, you can see that several features will be used to classify the quality of wine, many of them are chemical, so we need to have a basic understanding of such chemicals.

- **volatile acidity :** Volatile acidity *is the* gaseous acids present in wine.

- **fixed acidity :** Primary **fixed acids** found in wine are **tartaric**, **succinic**, **citric**, and **malic**
- **residual sugar :** Amount of sugar left after fermentation.
- **citric acid :** It is weak organic acid, found in citrus fruits naturally.
- **chlorides :** Amount of salt present in wine.
- **free sulfur dioxide :** So2 is used for prevention of wine by oxidation and microbial spoilage.
- **total sulfur dioxide**
- **pH :** In wine pH is used for checking acidity
- **density**
- **sulphates** : Added sulfites preserve freshness and protect **wine** from oxidation, and bacteria.
- **alcohol :** Percent of alcohol present in wine.

Rather than chemical features, you can see that there is one feature named **Type** it contains the types of wine we here discuss on **red** and **white** wine, the percent of red wine is greater than white.

```
df.describe(include='all')
```

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
count	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000
mean	8.319637	0.527821	0.270976	2.538806	0.087467	15.874922	46.467792	0.996747	3.311113	0.658149	10.422983	5.636023
std	1.741096	0.179060	0.194801	1.409928	0.047065	10.460157	32.895324	0.001887	0.154386	0.169507	1.065668	0.807569
min	4.600000	0.120000	0.000000	0.900000	0.012000	1.000000	6.000000	0.990070	2.740000	0.330000	8.400000	3.000000
25%	7.100000	0.390000	0.090000	1.900000	0.070000	7.000000	22.000000	0.995600	3.210000	0.550000	9.500000	5.000000
50%	7.900000	0.520000	0.260000	2.200000	0.079000	14.000000	38.000000	0.996750	3.310000	0.620000	10.200000	6.000000
75%	9.200000	0.640000	0.420000	2.600000	0.090000	21.000000	62.000000	0.997835	3.400000	0.730000	11.100000	6.000000
max	15.900000	1.580000	1.000000	15.500000	0.611000	72.000000	289.000000	1.003690	4.010000	2.000000	14.900000	8.000000

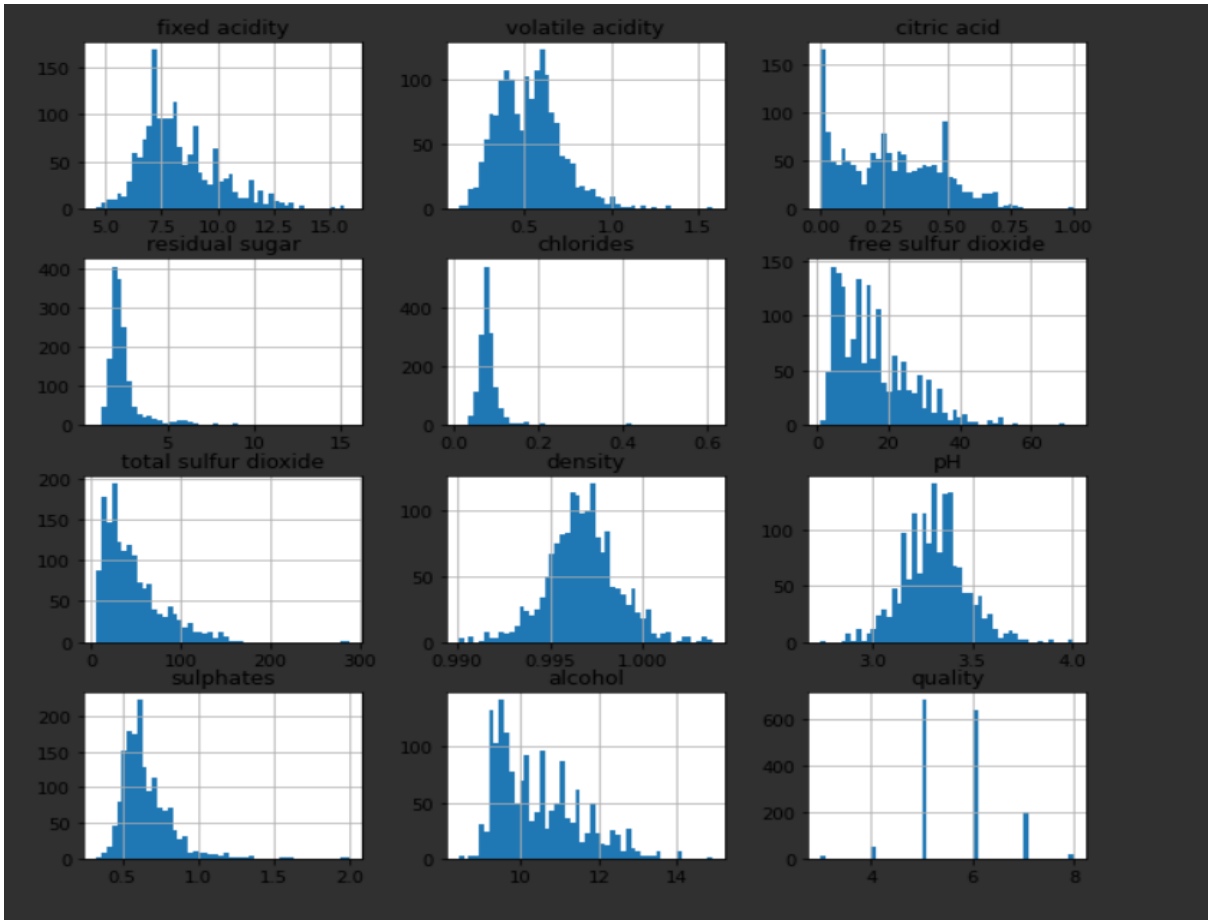
VISUALIZATION

We know that the “image speaks everything” here the visualization came into the work, we use visualization for explaining the data. In other words, we can say that it is a graphic representation of data that is used to find useful information.

Histogram

```
df.hist(figsize=(10,10),bins=50)
plt.show()
```

Output:

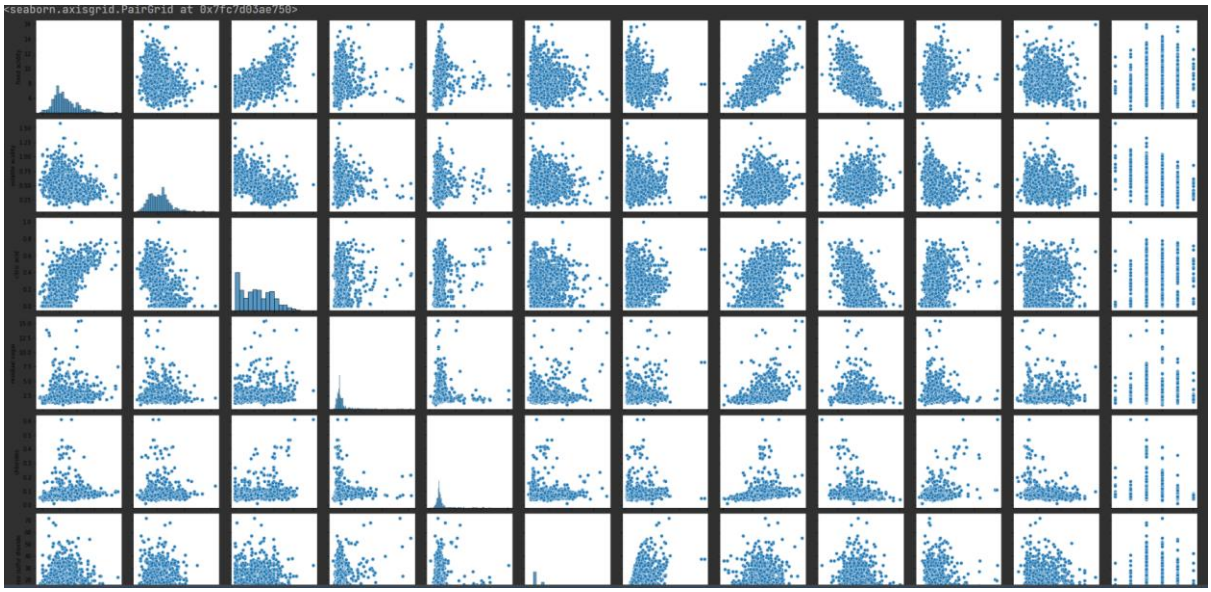


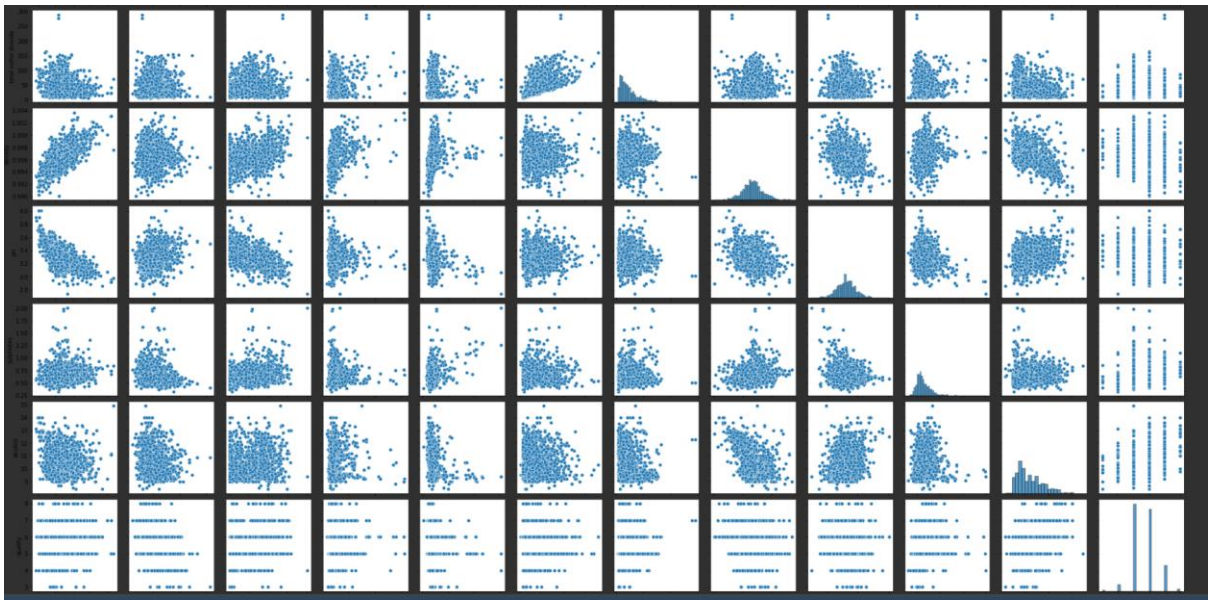
plot the graph in which we check what value of alcohol can able to make changes in quality.

Pair Plot:

```
sns.pairplot(df)
```

Output:





When we performing any machine learning operations then we have to study the data features deep, there are many ways by which we can differentiate each of the features easily. Now, we will perform a correlation on the data to see how many features are there they correlated to each other.

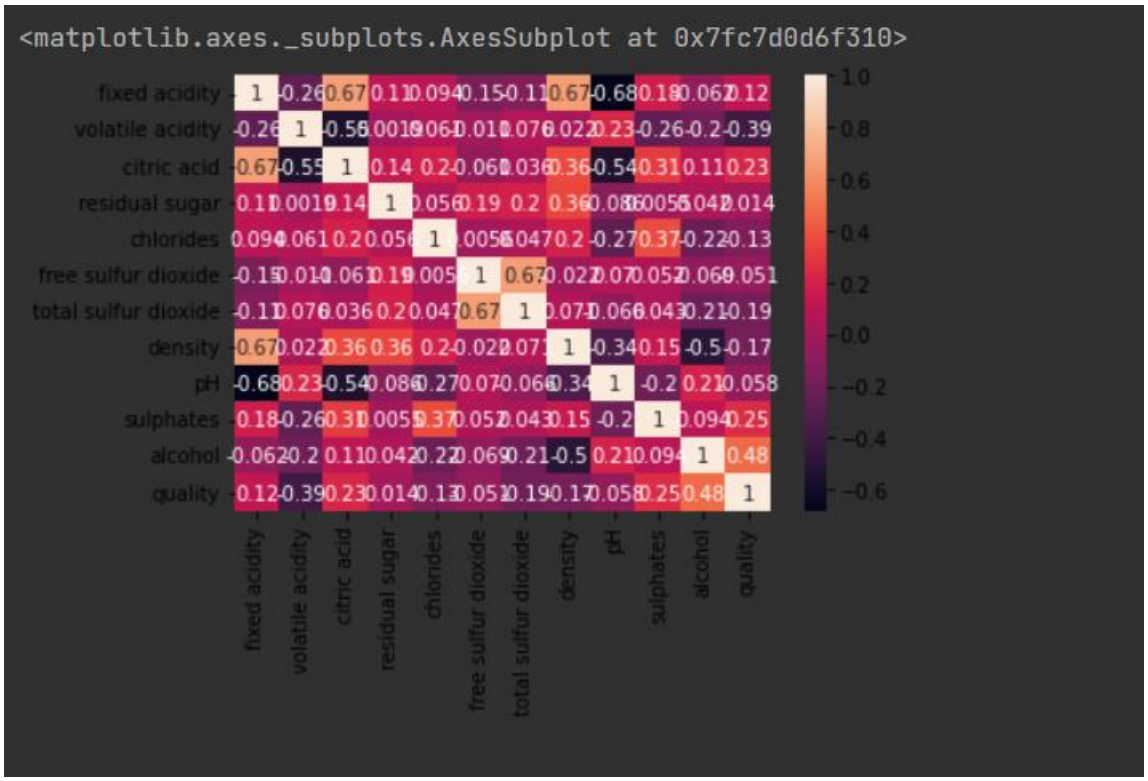
CORRELATION:-

For checking correlation we use a statistical method that finds the bonding and relationship between two features.

Heatmap for expressing correlation

```
corr = df.corr()
sns.heatmap(corr,annot=True)
```

Output:



Now, we have to find those features that are fully correlated to each other by this we reduce the number of features from the data.

If you think that why we have to discard those correlated, because relationship among them is equal they equally impact on model accuracy so, we delete one of them.

Normalization

We do normalization on numerical data because our data is unbalanced it means the difference between the variable values is high so we convert them into 1 and 0.

```
#importing module
from sklearn.preprocessing import MinMaxScaler
# creating normalization object
norm = MinMaxScaler()
# fit data
norm_fit = norm.fit(x_train)
new_xtrain = norm_fit.transform(x_train)
new_xtest = norm_fit.transform(x_test)
# display values
print(new_xtrain)
```

HANDLE NULL VALUES

```
new_df.isnull().sum()
```

In the dataset, there is so much notice data present, which will affect the accuracy of our ML model. In machine learning, there are many ways to handle null or missing values. Now, we will use them to handle our unorganized data.

Finding Null Values

```
print(df.isna().sum())
```

```
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
```

RANDOM FOREST CLASSIFIER

This is the last step where we apply any suitable model which will give more accuracy, here we will use ***RandomForestClassifier*** because it was the only ML model that gives the 89% accuracy which was considered as the best accuracy.

Using Random Forest:

```
from sklearn.ensemble import RandomForestClassifier
model2 = RandomForestClassifier(random_state=1)
model2.fit(X_train, Y_train)
y_pred2 = model2.predict(X_test)

from sklearn.metrics import accuracy_score
print("Accuracy Score:",accuracy_score(Y_test,y_pred2))
```

```
Accuracy Score: 0.89375
```

Results:

```
results = pd.DataFrame({
    'Model': ['Logistic Regression','KNN','Decision Tree' ,'Random Forest','Xgboost'],
    'Score': [0.870,0.872,0.864,0.893,0.879]})

result_df = results.sort_values(by='Score', ascending=False)
result_df = result_df.set_index('Score')
result_df
```

Score	Model
0.893	Random Forest
0.879	Xgboost
0.872	KNN
0.870	Logistic Regression
0.864	Decision Tree

Work Division

Bijja Thimothy Lipsika

- Project code
- Data Set
- Project Implementation

Nagam Haritha

- Project code
- Helps In Finding of Data Set
- Report

Thank You!