Prediction of Wine Quality based on the given dataset for Red and White Wine

Name: Arunesh Kumar Registration Number: 185001024

Problem Statement: -

In recent years there is a modest increase in the wine consumption as it has been found that wine consumption has a positive correlation to the heart rate variability. With the increase in the consumption wine industries are looking for alternatives to produce good quality wine at less cost. Different wines have different purposes.

Although most of the chemicals are same for different type of wine based on the chemical tests, the quantity of each chemical have different level of concentration for different type of wine. These days it is really important to classify different wine for quality assurance. In the past due to lack of technological resources it become difficult for most of the industries to classify the wines based on the chemical analyses as it takes lot of time and also need more money.

These days with the advent of the machine learning techniques it is possible to classify the wines as well as it is possible to figure out the importance of each chemical analyses parameters in the wine and which one to ignore for reduction of cost. The performance comparison with different feature sets will also help to classify it in a more distinctive way. In this paper machine learning approach is proposed by considering based feature selection considering the classifiers, linear classifiers and probabilistic classifiers to predict the quality in red and white wine. [1]

Survey of Research Papers based on wine quality: -

In the first paper [1], the author uses a data set similar to the one used here. The author proposes a data mining approach to predict human wine taste preferences that is based on easily available analytical tests at the certification step. Three regression techniques were applied, under a computationally efficient procedure that performs simultaneous variable and model selection. The support vector machine achieved promising results, outperforming the multiple regression and neural network methods.

Such model is useful to support the oenologist wine tasting evaluations and improve wine production. In this quality prediction testing is done on the 20 percent of the data and the training is done on the 80 percent of the data. They achieved various accuracies for the techniques they used and for their data set random forest achieved an accuracy of 1. While rest were at around 50 to 70 percent for their dataset.

In another paper [2], The author uses another same data set based on wine quality. The paper explores the usage of ML techniques such as Linear Regression, neural Network and Support Vector Machines for wine quality in 2 ways.

First they determine the dependency of the target variable on independent variables and predict the value for the target variable. They used Linear regression to determine the dependency. This paper concludes with stating that better prediction can be made if selected variables or attributes are being considered rather than all features.

Scope of the project: -

- ➔ To determine which ML prediction algorithm would be best suited for the given datasets
- ➔ perform data visualizations
- ➔ hypothesis testing on the output variable
- ➔ a set of features to get accurate prediction

Dataset Description: -

- ⇨ There are a total of 2 types of wine quality csv files, winequailty-white.csv and winequailty-red.csv each having the following structure: -
    1. **Winequailty-white.csv:**
        a. Total rows: 4998
        b. Total columns: 12
    2. **Winequality-red.csv:**
        a. Total rows: 1599
        b. Total columns: 12
- ⇨ The dataset was obtained from https://archive.ics.uci.edu/ml/datasets/wine+quality [3].
- ⇨ The two datasets are related to red and white variants of the Portuguese "Vinho Verde" wine.
- ⇨ Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available (e.g. there is no data about grape types, wine brand, wine selling price, etc.).
- ⇨ Attribute Description: -

    1. Alcohol: the amount of alcohol in wine

    2. Volatile acidity: are high acetic acid in wine which leads to an unpleasant vinegar taste

    3. Sulphates: a wine additive that contributes to SO2 levels and acts as an antimicrobial and antioxidant

    4. Citric Acid: acts as a preservative to increase acidity (small quantities add freshness and flavor to wines)

    5. Total Sulfur Dioxide: is the amount of free + bound forms of SO2

    6. Density: sweeter wines have a higher density

    7. Chlorides: the amount of salt in the wine

8. Fixed acidity: are non-volatile acids that do not evaporate readily

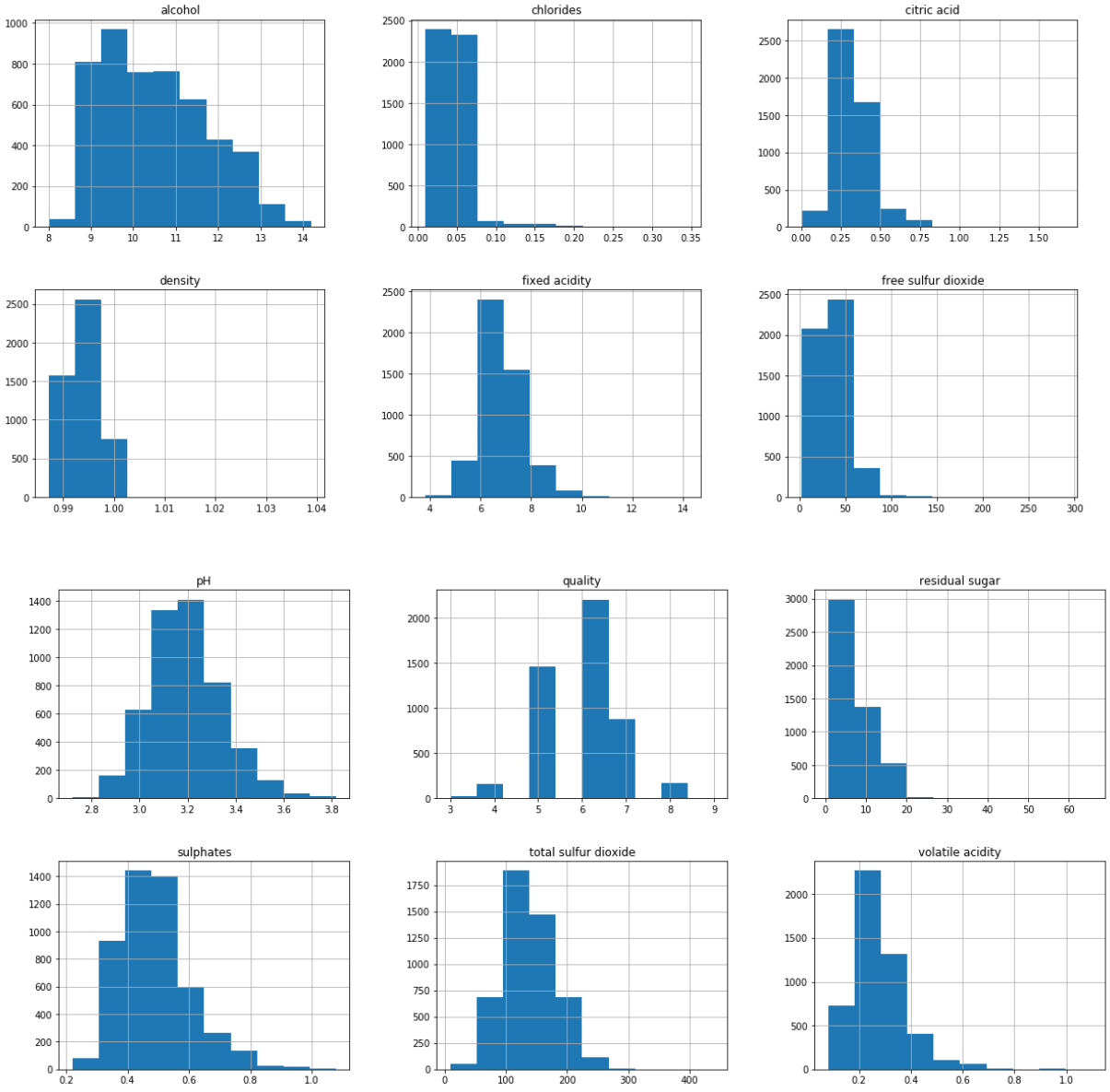
9. pH: the level of acidity

10. Free Sulfur Dioxide: it prevents microbial growth and the oxidation of wine

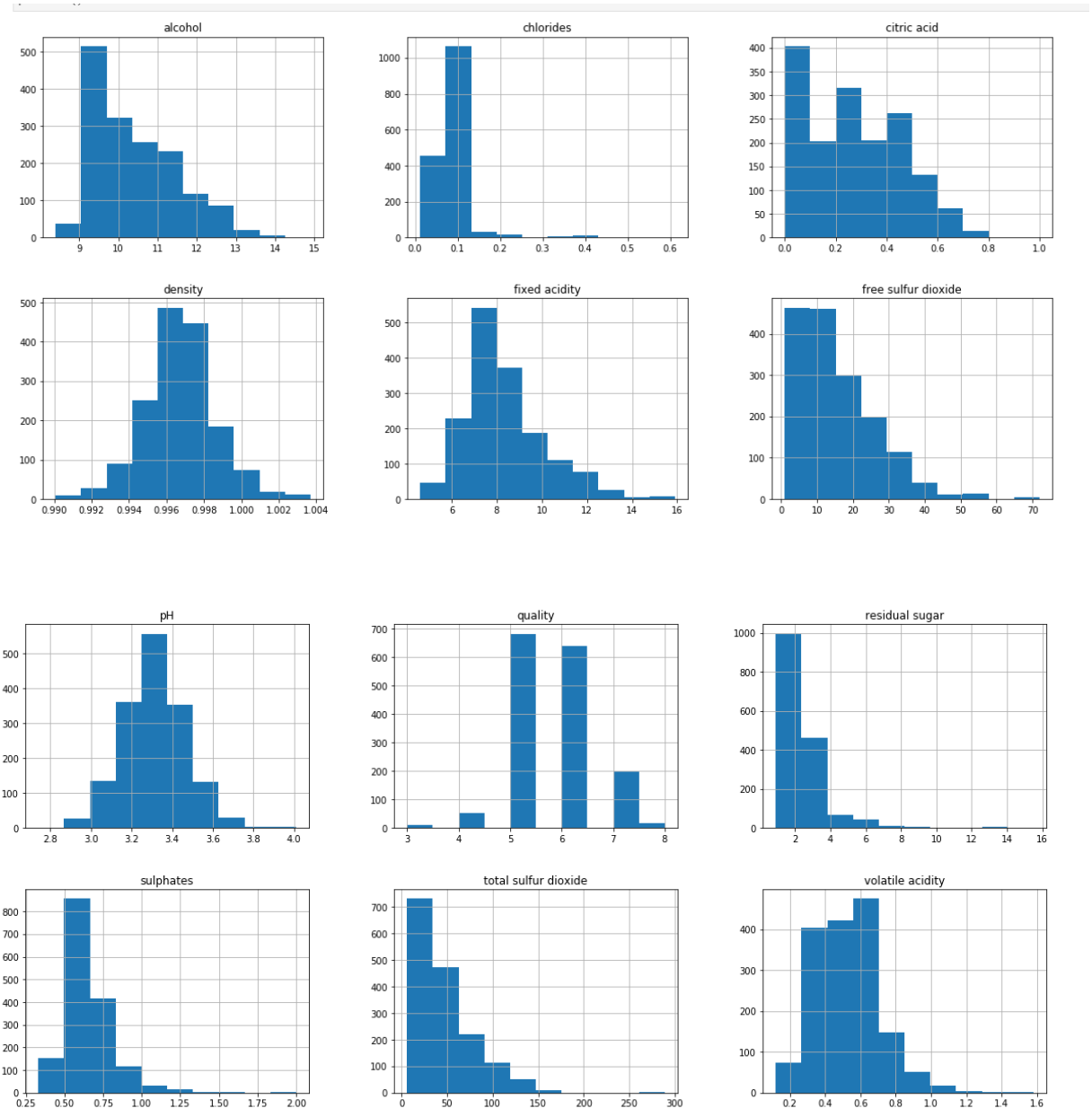
11. Residual sugar: is the amount of sugar remaining after fermentation stops. The key is to have a perfect balance between — sweetness and sourness (wines > 45g/ltrs are sweet)

Data Visualizations:

- ⇨ **The histogram of all variables in datasets: -**
  - **For white-wine-quality: -**

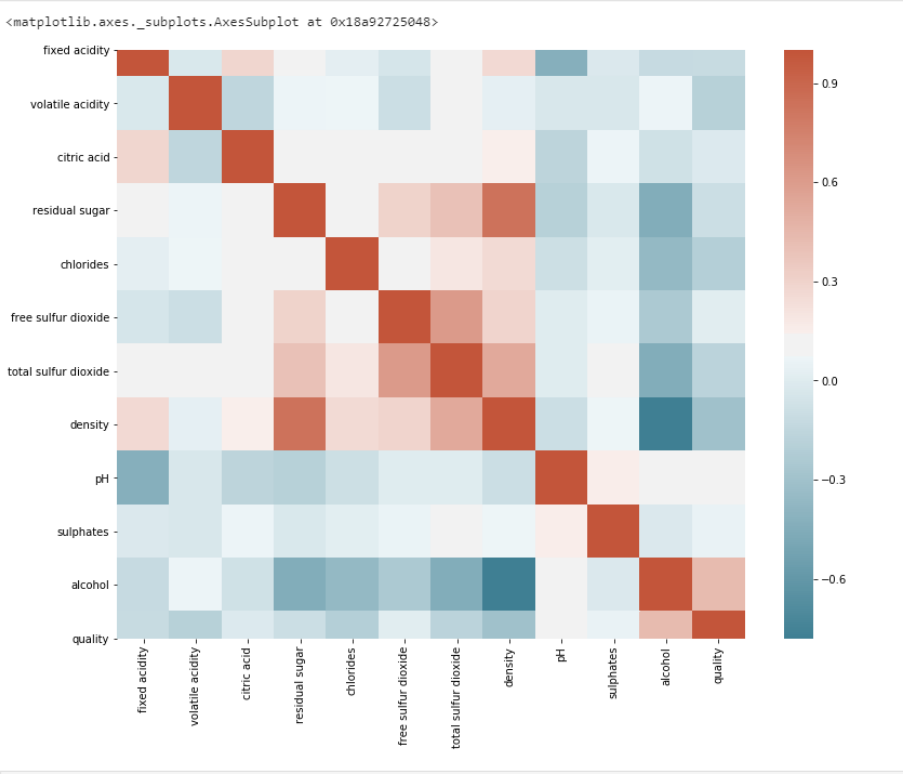


- **For red-wine-quality: -**

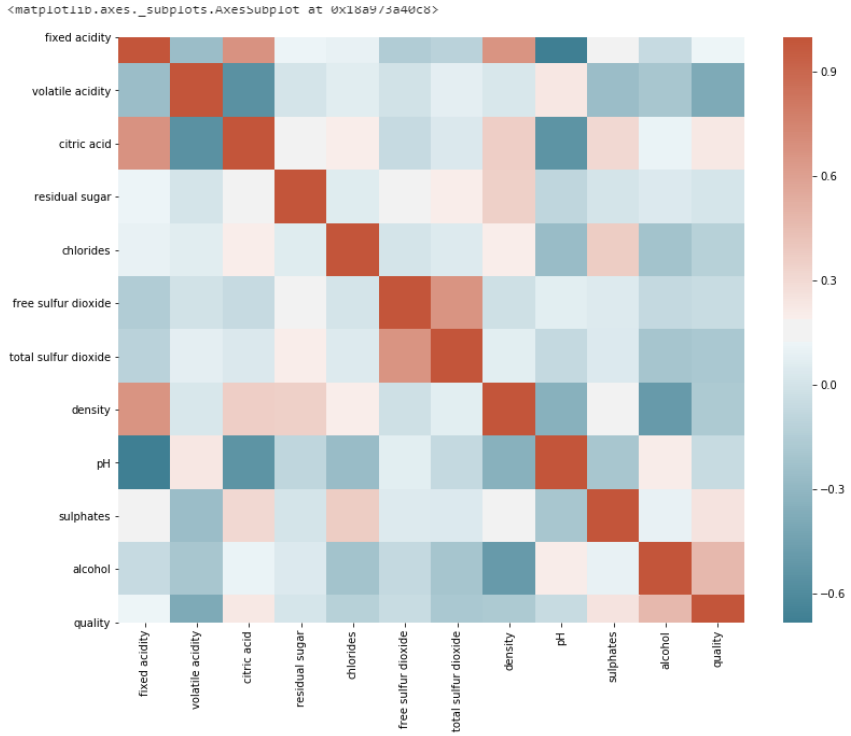


⇨ **The correlation matrix to determine the most impacting feature in the datasets: -**

- **For white-wine-quality: -**



- **For red-wine-quality: -**



⇨ **The dependency of which feature effects the quality the most: -**

- **For white-wine-quality: -**

```
corr_white['quality'].sort_values(ascending=False)
```

```
quality                1.000000
alcohol                0.435575
pH                     0.099427
sulphates              0.053678
free sulfur dioxide    0.008158
citric acid           -0.009209
residual sugar        -0.097577
fixed acidity         -0.113663
total sulfur dioxide  -0.174737
volatile acidity      -0.194723
chlorides             -0.209934
density               -0.307123
Name: quality, dtype: float64
```

- **For red-wine-quality: -**

```
corr_red['quality'].sort_values(ascending=False)
```

```
quality                1.000000
alcohol                0.476166
sulphates              0.251397
citric acid            0.226373
fixed acidity          0.124052
residual sugar         0.013732
free sulfur dioxide   -0.050656
pH                    -0.057731
chlorides             -0.128907
density               -0.174919
total sulfur dioxide  -0.185100
volatile acidity      -0.390558
Name: quality, dtype: float64
```

Proposed Algorithm to be used: -

We are going to perform the given below algorithms and comparing with which one has the highest accuracy: -

1. **Logistic Regression**
2. **SGD Classifier (Stochastic Gradient Decent Classifier)**
3. **Gradient Boosting Classifier**
4. **Random Forest**
5. **Support Vector Classifier (SVC)**
6. **Decision Tree Classifier**

Tools used to perform the algorithm: -

1. **Logistic Regression:**

```
from sklearn.linear_model import LogisticRegression
```

2. **SGD Classifier (Stochastic Gradient Decent Classifier):**

```
from sklearn.linear_model import SGDClassifier
```

3. **Gradient Boosting Classifier:**

```
from sklearn.ensemble import GradientBoostingClassifier
```

4. **Random Forest:**

```
from sklearn.ensemble import RandomForestClassifier
```

5. **Support Vector Classifier (SVC):**

```
from sklearn.svm import SVC
```

6. **Decision Tree Classifier:**

```
from sklearn.tree import DecisionTreeClassifier
```

Other important tools: -

```
⇨ import seaborn as sns
⇨ import pandas as pd
```

```
⇨ import matplotlib.pyplot as plt
⇨ import plotly.express as px
⇨ from scipy import stats
⇨ import plotly
⇨ import plotly.offline as py
⇨ import plotly.graph_objs as go
⇨ from sklearn.preprocessing import LabelEncoder
⇨ from sklearn.metrics import classification_report
⇨ from sklearn.model_selection import train_test_split
⇨ from sklearn.metrics import confusion_matrix
```

Performance Metrics: -

- There are four ways to check if the predictions are right or wrong:
  - **TN / True Negative**: the case was negative and predicted negative
  - **TP / True Positive**: the case was positive and predicted positive
  - **FN / False Negative**: the case was positive but predicted negative
  - **FP / False Positive**: the case was negative but predicted positive
- **Precision** — What percent of your predictions were correct?
  - Precision is the ability of a classifier not to label an instance positive that is actually negative.
  - For each class, it is defined as the ratio of true positives to the sum of a true positive and false positive.
  - Precision: - Accuracy of positive predictions.
  - Precision = TP/(TP + FP)
- **Recall** — What percent of the positive cases did you catch?
  - Recall is the ability of a classifier to find all positive instances. For each class it is defined as the ratio of true positives to the sum of true positives and false negatives.
  - Fraction of positives that were correctly identified.
  - Recall = TP/(TP+FN)
- **F1 score** — What percent of positive predictions were correct?
  - The F1 score is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst is 0.0.
  - F1 scores are lower than accuracy measures as they embed precision and recall into their computation.
  - As a rule of thumb, the weighted average of F1 should be used to compare classifier models, not global accuracy.
  - F1 Score = 2*(Recall * Precision) / (Recall + Precision)

- **Support**
  - Support is the number of actual occurrences of the class in the specified dataset.
  - Imbalanced support in the training data may indicate structural weaknesses in the reported scores of the classifier and could indicate the need for stratified sampling or rebalancing.
  - Support doesn't change between models but instead diagnoses the evaluation process.
- **Accuracy**
  - Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.
  - It is a great measure but only when you have symmetric datasets where values of false positive and false negatives are almost same.
  - Accuracy = TP+TN/TP+FP+FN+TN

Results: -

- The results of classification_report() for white wine are as follows: -

1. Logistic Regression: -

	precision	recall	f1-score	support
0	0.81	0.97	0.89	955
1	0.70	0.20	0.32	270
accuracy			0.80	1225
macro avg	0.75	0.59	0.60	1225
weighted avg	0.79	0.80	0.76	1225

2. SGD Classifier: -

	precision	recall	f1-score	support
0	0.84	0.92	0.88	955
1	0.57	0.38	0.46	270
accuracy			0.80	1225
macro avg	0.70	0.65	0.67	1225
weighted avg	0.78	0.80	0.78	1225

3. Gradient Boosting Classifier: -

	precision	recall	f1-score	support
0	0.85	0.95	0.90	955
1	0.69	0.40	0.51	270
accuracy			0.83	1225
macro avg	0.77	0.68	0.70	1225
weighted avg	0.81	0.83	0.81	1225

4. Random Forest: -

	precision	recall	f1-score	support
0	0.90	0.95	0.92	955
1	0.77	0.64	0.70	270
accuracy			0.88	1225
macro avg	0.84	0.79	0.81	1225
weighted avg	0.87	0.88	0.87	1225

5. Support Vector Classifier: -

	precision	recall	f1-score	support
0	0.81	0.96	0.88	955
1	0.62	0.21	0.31	270
accuracy			0.80	1225
macro avg	0.72	0.59	0.60	1225
weighted avg	0.77	0.80	0.76	1225

6. Decision Tree Classifier: -

	precision	recall	f1-score	support
0	0.91	0.87	0.89	955
1	0.60	0.70	0.64	270
accuracy			0.83	1225
macro avg	0.75	0.78	0.77	1225
weighted avg	0.84	0.83	0.83	1225

⇨ The results of classification_report() for red wine are as follows: -

1. Logistic Regression: -

	precision	recall	f1-score	support
0	0.86	0.99	0.92	341
1	0.43	0.05	0.09	59
accuracy			0.85	400
macro avg	0.64	0.52	0.50	400
weighted avg	0.79	0.85	0.80	400

2. SGD Classifier: -

	precision	recall	f1-score	support
0	0.86	0.99	0.92	341
1	0.62	0.08	0.15	59
accuracy			0.86	400
macro avg	0.74	0.54	0.54	400
weighted avg	0.83	0.86	0.81	400

3. Gradient Boosting Classifier: -

	precision	recall	f1-score	support
0	0.90	0.96	0.93	341
1	0.63	0.41	0.49	59
accuracy			0.88	400
macro avg	0.77	0.68	0.71	400
weighted avg	0.86	0.88	0.87	400

4. Random Forest: -

	precision	recall	f1-score	support
0	0.92	0.97	0.95	341
1	0.77	0.51	0.61	59
accuracy			0.91	400
macro avg	0.84	0.74	0.78	400
weighted avg	0.90	0.91	0.90	400

5. Support Vector Classifier: -

	precision	recall	f1-score	support
0	0.87	0.98	0.92	341
1	0.60	0.15	0.24	59
accuracy			0.86	400
macro avg	0.74	0.57	0.58	400
weighted avg	0.83	0.86	0.82	400

6. Decision Tree Classifier: -

	precision	recall	f1-score	support
0	0.92	0.94	0.93	341
1	0.61	0.56	0.58	59
accuracy			0.88	400
macro avg	0.77	0.75	0.76	400
weighted avg	0.88	0.88	0.88	400

⇨ **The results are as follows for each classifier in both white and red wine: -**

Accuracy		
Algorithm Used	**White wine**	**Red Wine**
Logistic Regression	80	85
SGD Classifier	80	86
Gradient Boosting Classifier	83	88
Random Forest	88	91
Support Vector Classifier	80	86
Decision Tree Classifier	83	88

Conclusion: -

For the given scope the following conclusions have been drawn: -

➔ The following are the better ML algorithm for the given datasets: -

- Random Forest
- Gradient Boosting Classifier
- Decision Tree Classifier

- ➔ We have performed the data visualizations on both datasets which have been useful in determining the useful features.
- ➔ The useful features that are determined in both datasets are as follows: -
    - White-wine-quality.csv:
        - free sulfur dioxide
        - pH
        - Sulphates
        - Alcohol
    - Red-wine-quality.csv:
        - Alcohol
        - Sulphates
        - fixed acidity
        - citric acid

References: -

[1] Nikita Sharma, "Quality Prediction of Red Wine based on Different Feature Sets Using Machine Learning Techniques", International Journal of Science and Research (IJSR), https://www.ijsr.net/search_index_results_paperid.php?id=SR20718002904, Volume 9 Issue 7, July 2020, 1358 - 1366

[2] Gupta, Yogesh. (2018). Selection of important features and predicting wine quality using machine learning techniques. Procedia Computer Science. 125. 305-312. 10.1016/j.procs.2017.12.041.

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