B.Tech - Computer and Communication Engineering

Department of Electronics and Communication Engineering

Amrita School of Engineering, Coimbatore- 641112

AMRITA
VISHWA VIDYAPEETHAM
DEEMED TO BE UNIVERSITY

Wine Quality Prediction Model using Machine Learning

19CCE213 Machine Learning and Artificial Intelligence

End Semester Project

Class and Semester: B.Tech CCE, Fourth Semester

Roll No: CB.EN.U4CCE20013
Name: Damisetty Venkata Sai Revanth

Roll No: CB.EN.U4CCE20026
Name: Karthikeyan Saravanan

Roll No: CB.EN.U4CCE20012
Name: Chilaka Sohith Reddy

Roll No: CB.EN.U4CCE20057
Name: Siva Dhanush Kosuri

Ms. G Suguna
Assistant Professor
Department of Electronics and Communication Engineering
Amrita School of Engineering, Coimbatore- 641112

3 August 2022

Abstract

The standard of a wine is salient for the customers as well as the wine production industries. The conventional course of action of assessing wine calibre is sluggish. At the moment, predictive analysis are principal mechanism to restore the endeavour. In this instance, there are assorted attributes to anticipate the wine quality but the complete trait will not be applicable for finer forecast. Our project pivots on what wine aspect are dominant to get the encouraging result. With the aim of classing model and evaluation of the relevant features, we cast-off four algorithms namely support vector machine (SVM), K Nearest Neighbour Classifier, Decision Tree and Random Forest Classifier. In this research, we utilised a wine quality dataset comprising of the red wine and white wine. To assess the quality importance we utilised the Pearson coefficient correlation and performance measurement matrices such as recall, accuracy, precision, and f1 score for comparison of the machine learning algorithm. Finally, we achieved the results for the Support Vector Machine (SVM) algorithm, K-Nearest Neighbour, Decision Tree and Random Forest Classifier for the dataset comprising red wine and white wine.

Keywords— Classification, Support Vector Machine, K-Nearest Neighbour, Decision Tree, Random Forest Classifier, AUC-ROC Curve.

Motivation

The principal motivation following this examination is to envisage wine calibre depending on physicochemical knowledge using Artificial Intelligence and Machine Learning. The vital factor in red wine certification and quality assessment is physicochemical tests, which are laboratory-based and consider factors like acidity, pH level, sugar, and other chemical properties. The project motivated us to try different feature selection algorithm as well as different classifiers to compare the performance metrics. This project aims to determine which features are the best quality red and white wine indicators and generate insights into each of these factors to our model's red and white wine quality. Knowing how each variable will impact the red and white wine quality will help producers, distributors, and businesses in the red wine industry better assess their production, distribution, and pricing strategy.

Research Paper Reviews

- **Kumar et al. (2020)** have used prediction of red wine quality using its various attributes and for the prediction, they used random forest, support vector machine, and naive Bayes techniques (Kumar et al., 2020). They have calculated the performance measurement such as precision, recall, f1-score, accuracy, specificity, and misclassification error. Among these three techniques, they achieved the best result from the support vector machine as compare to the random forest and naive Bayes techniques. They achieved the accuracy of the support vector machine technique is 67.25%.

- **Gupta, (2018)** has used important features from red wine and white wine quality using various machine learning algorithms such as linear regression, neural network, and support vector machine techniques. They used two ways to determine the wine quality. Firstly the dependency of the target variable on the independent variable and secondly predicting the value of the target variable and conclusion that all features are not necessary for the prediction instead of selecting only necessary features to predict the wine quality (Gupta, 2018).

- **Dahal et al., (2021)** has predicted the wine quality based on the various parameters by applying various machine learning models such as rigid regression, support vector machine, gradient boosting regressor, and multi-layer artificial neural network. They compare the performance of the models to predict wine quality and from their analysis, they found gradient boosting regressor is the best model to other model performances with the MSE, R, and MAPE of 0.3741, 0.6057, and 0.0873 respectively(Dahal et al., 2021).

- **Er, and Atasoy, (2016)** has proposed the method to classify the quality of the red wine and white wine using three machine learning algorithm such as k-nearest-neighborhood, random forest, and support vector machine. They used principal component analysis for the feature selection and they have achieved the best result using the random forest algorithm (Er, 2016).

- **Lee et al., (2015)** has proposed a method decision tree-based to predict the wine quality and compare their approach using three machine learning algorithm such as support vector machine, multi-layer perceptron, and BayesNet. They found their proposed method is better compared to other stated methods (Lee et al., 2015).

- **P. Appalasamy et al., (2012)** have predicted the wine quality based on the physiochemical data. They used both red wine and white wine datasets and applied the decision tree and naive Bayes algorithms. They compare the results of these two algorithms and conclude that the classification approach can help to improve the wine quality during production (P. Appalasamy et al., 2012).

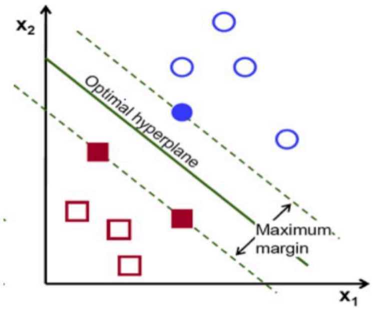
Classification Descriptions

A wide range of machine learning algorithms is available for the learning process. The description of the classification algorithms used in wine quality prediction and related work is stated as:

- **Support Vector Machine**

The support vector machine (SVM) is the most popular and most widely used machine learning algorithm. It is a supervised learning model that can perform classification and regression tasks. However, it is primarily used for classification problems in machine learning.

The SVM algorithm aims to create the best line or decision boundary that can separate n-dimensional space into classes. So we can put the new data points easily in the correct groups. This best decision boundary is called a hyperplane.



The support vector machine selects the extreme data points that helping to create the hyperplane. In Figure 1, two different groups are classified by using the decision boundary or hyperplane:

The SVM model is used for both non-linear and linear data. It uses a nonlinear mapping to convert the main preparing information into a higher measurement. The model searches for the linear optimum splitting hyperplane in this new measurement. A hyperplane can split the data into two classes with an appropriate nonlinear mapping to suitably high measurements and for the finding, this hyperplane SVM uses the support vectors and edges (J. Han et al., 2012). The SVM model is a representation of the models as a point in space, the different classes are isolated by the gap to mapped with the aim that instances are wide as would be careful. The model can perform out a nonlinear form of classification (Kumar et al., 2020).

- **K-Nearest Neighbour**

The K-Nearest Neighbours algorithm, also known as KNN is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.

- **Decision Tree**

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Decision trees learn from data to approximate a sine curve with a set of if-then-else decision rules. The deeper the tree, the more complex the decision rules and the fitter the model.

- **Random Forest Classifier**

Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. The random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

Method and Approach

The performance measurement is calculated and evaluate the techniques to detect the effectiveness and efficiency of the model.

There are four ways to check the predictions are correct or incorrect:

- **True Positive:** Number of samples that are predicted to be positive which are truly positive.
- **False Positive:** Number of samples that are predicted to be positive which are truly negative.
- **False Negative:** Number of samples that are predicted to be negative which are truly positive.
- **True Negative:** Number of samples that are predicted to be negative which are truly negative.

Below listed techniques, we use for the evaluation of the model.

- **Accuracy** – Accuracy is defined as the ratio of correctly predicted observation to the total observation. The accuracy can be calculated easily by dividing the number of correct predictions by the total number of predictions.

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{False Positive} + \text{False Negative} + \text{True Negative}}$$

- **Precision** – Precision is defined as the ratio of correctly predicted positive observations to the total predicted positive observations.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

- **Recall** – Recall is defined as the ratio of correctly predicted positive observations to all observations in the actual class. The recall is also known as the True Positive rate calculated as,

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

- **F1 Score** – F1 score is the weighted average of precision and recall. The f1 score is used to measure the test accuracy of the model. F1 score is calculated by multiplying the recall and precision is divided by the recall and precision, and the result is calculated by multiplying two.

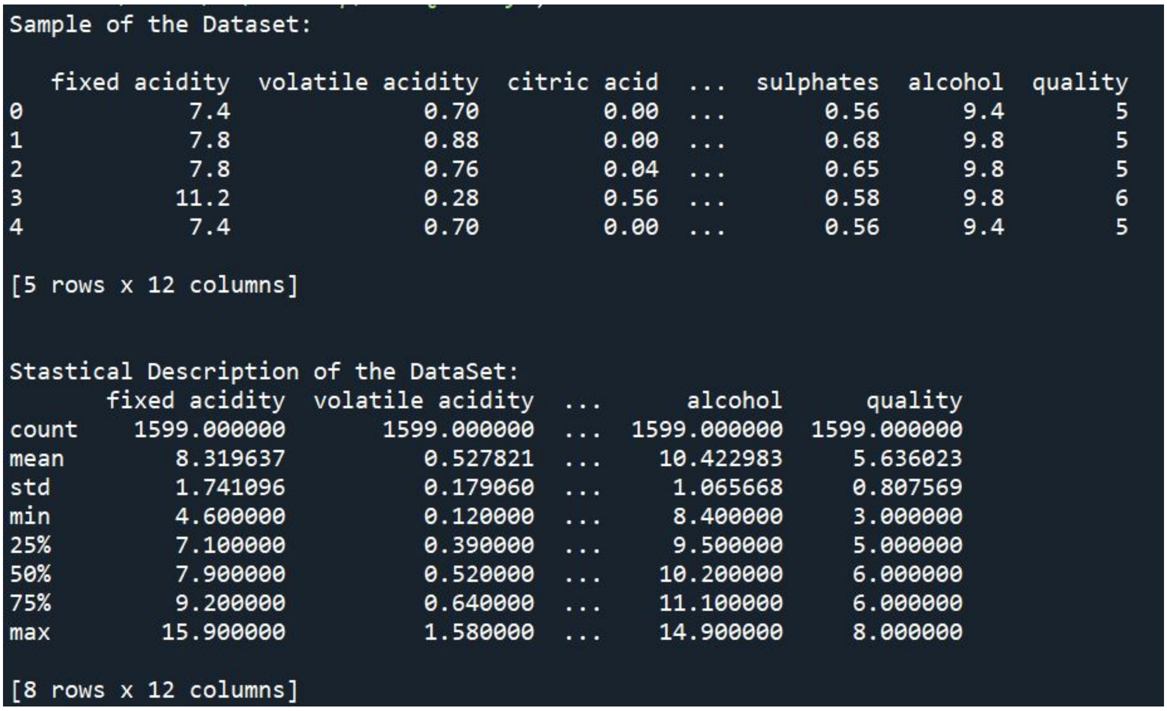
$$\text{F1 score} = 2 * \frac{\text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}}$$

- **AUC-ROC Curve -** AUC - ROC curve is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0 classes as 0 and 1 classes as 1.

Accuracy is the most widely used evaluation metric for most traditional applications. But the accuracy rate is not suitable for evaluating imbalanced data sets, because many experts have observed that for extremely skewed class distributions, the recall rate for minority classes is typically 0, which means that no classification rules are generated for the minority class. Using the terminology in information retrieval, the precision and recall of the minority categories are much lower than the majority class. Accuracy gives more weight to the majority class than to the minority class, this makes it challenging for the classifier to implement well in the minority class. For this purpose, additional metrics are coming into widespread usage (Guo et al., 2008).

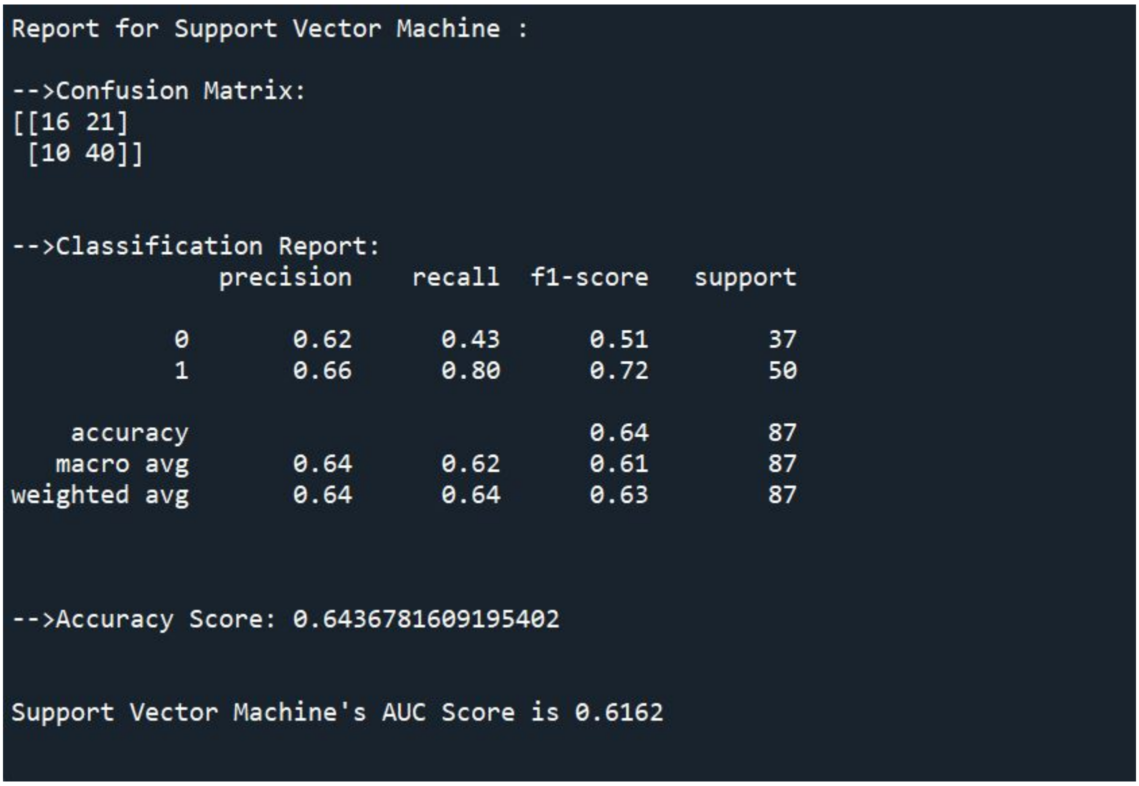
The F1 score is the popular evaluation metric for the imbalanced class problem (Estabrooks and Japkowicz, 2001). F1 score combines two matrices: precision and recall. Precision state how accurate the model was predicting a certain class and recall state that the opposite of the regrate misplaced instances which are misclassified. Since the multiple classes have multiple F1 scores. By using the unweighted mean of the F1 scores for our final scoring. We want our models to get optimized to classify instances that belong to the minority side, such as wine quality of 3, 8, or 9 equally well with the rest of the qualities that are represented in a larger number.

Results



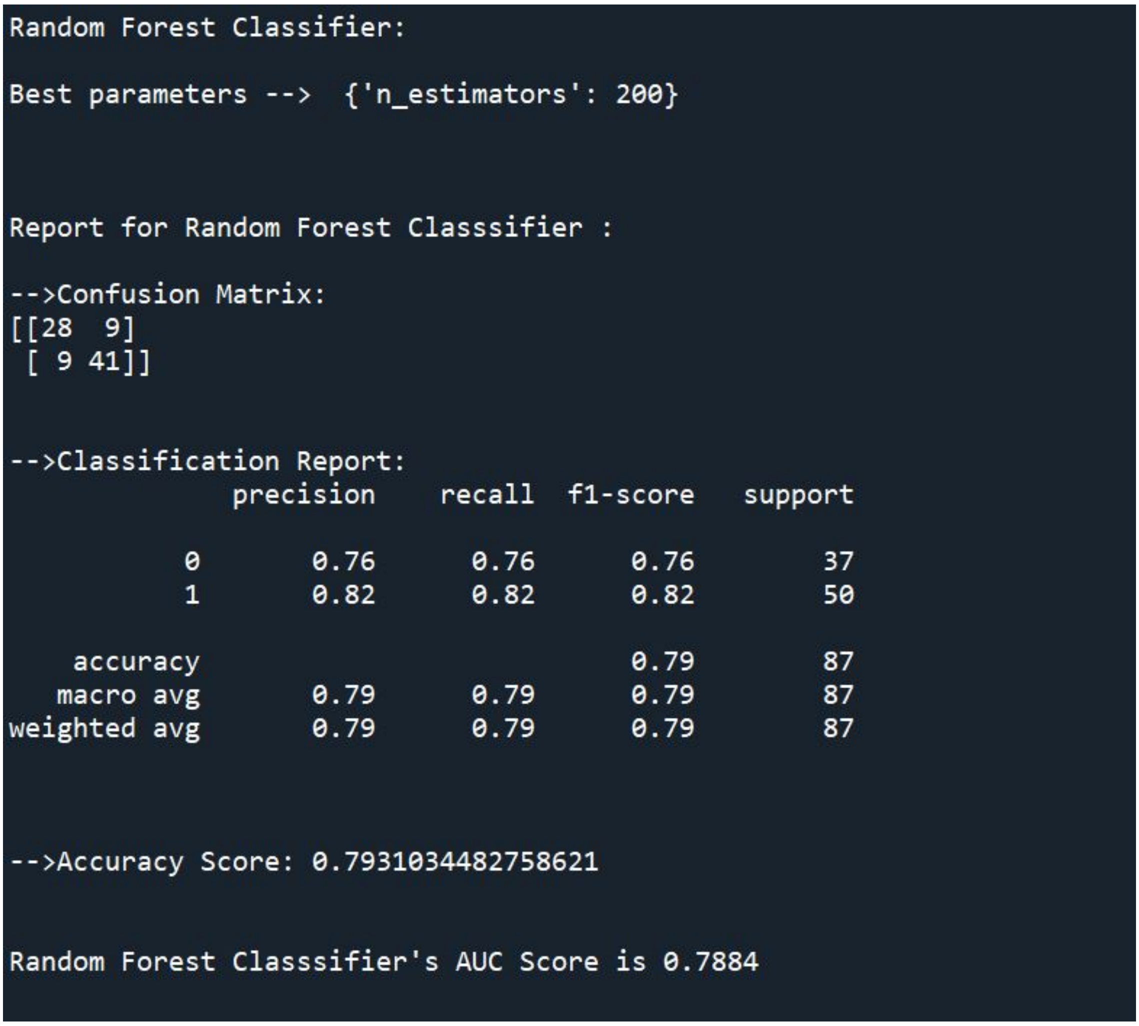
```
Sample of the Dataset:

   fixed acidity  volatile acidity  citric acid  ...  sulphates  alcohol  quality
0            7.4              0.70         0.00  ...       0.56      9.4        5
1            7.8              0.88         0.00  ...       0.68      9.8        5
2            7.8              0.76         0.04  ...       0.65      9.8        5
3           11.2              0.28         0.56  ...       0.58      9.8        6
4            7.4              0.70         0.00  ...       0.56      9.4        5

[5 rows x 12 columns]


Stastical Description of the DataSet:
       fixed acidity  volatile acidity  ...      alcohol      quality
count    1599.000000       1599.000000  ...  1599.000000  1599.000000
mean        8.319637          0.527821  ...    10.422983     5.636023
std         1.741096          0.179060  ...     1.065668     0.807569
min         4.600000          0.120000  ...     8.400000     3.000000
25%         7.100000          0.390000  ...     9.500000     5.000000
50%         7.900000          0.520000  ...    10.200000     6.000000
75%         9.200000          0.640000  ...    11.100000     6.000000
max        15.900000          1.580000  ...    14.900000     8.000000

[8 rows x 12 columns]
```

Figure 1: Red and White Wine dataset statistical analysis of the feature model.



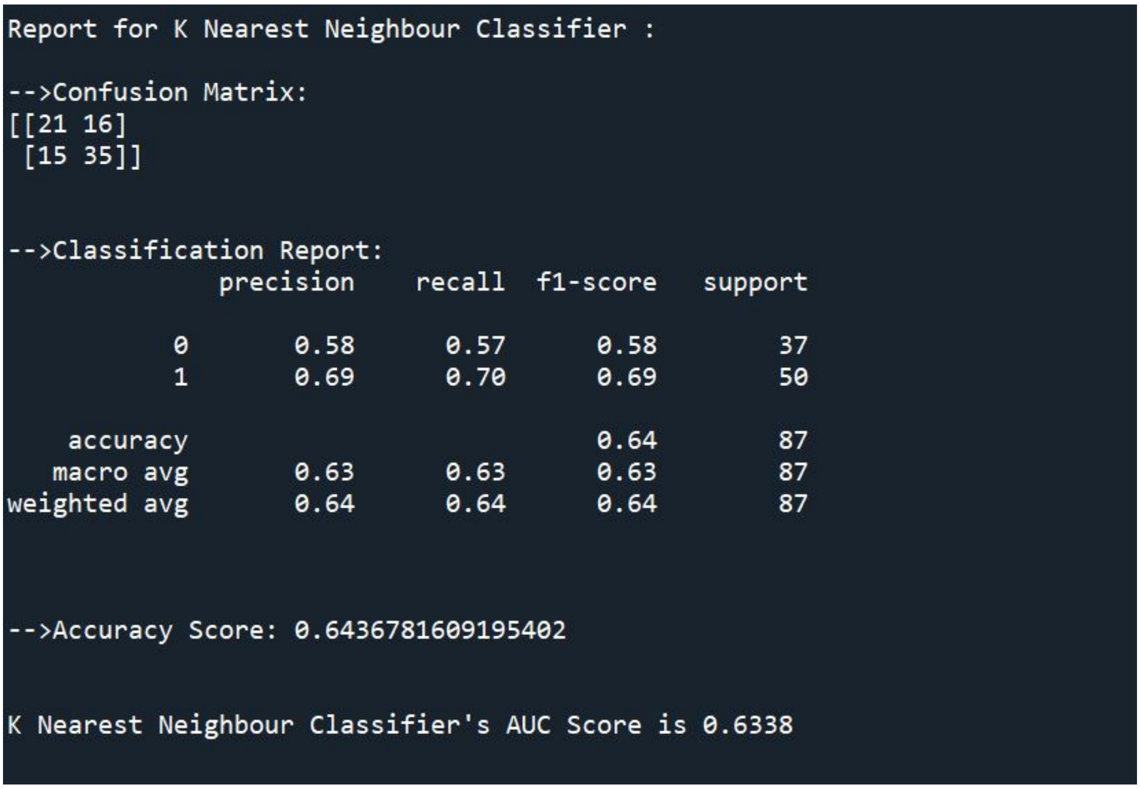
```
Report for Support Vector Machine :

-->Confusion Matrix:
[[16 21]
 [10 40]]


-->Classification Report:
              precision    recall  f1-score   support

           0       0.62      0.43      0.51        37
           1       0.66      0.80      0.72        50

    accuracy                           0.64        87
   macro avg       0.64      0.62      0.61        87
weighted avg       0.64      0.64      0.63        87



-->Accuracy Score: 0.6436781609195402


Support Vector Machine's AUC Score is 0.6162
```

Figure 2: Support Vector Machine (SVM) report of the feature model.



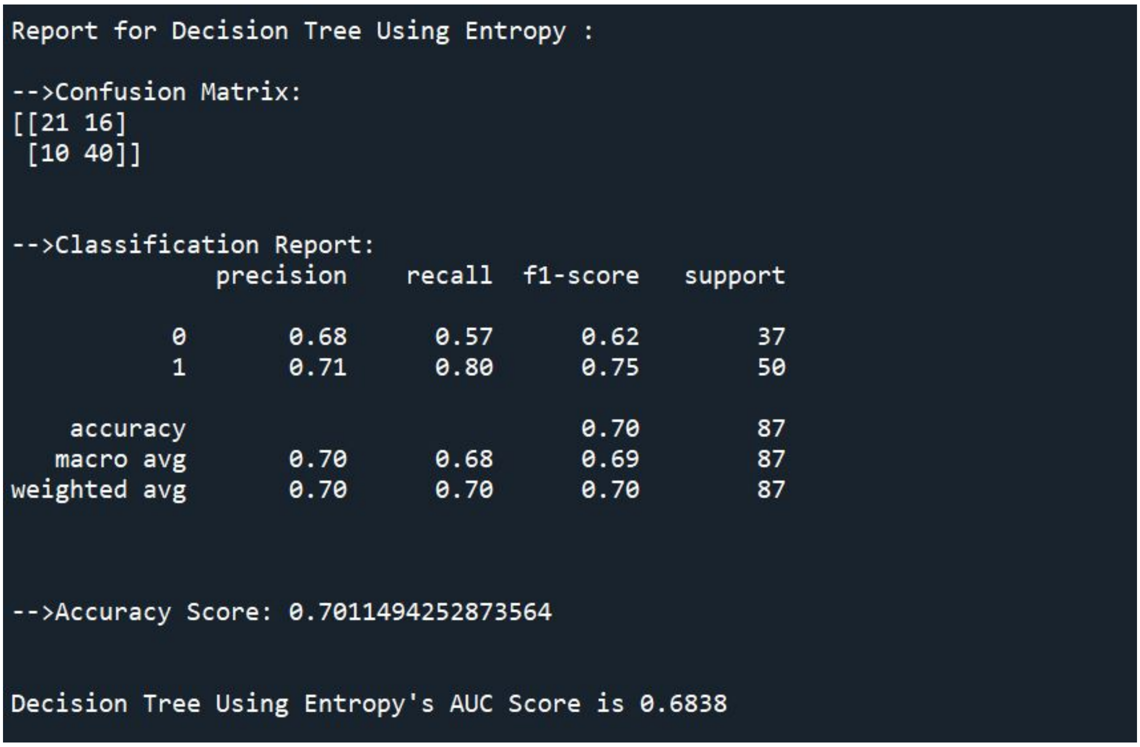
```
Random Forest Classifier:

Best parameters -->  {'n_estimators': 200}



Report for Random Forest Classsifier :

-->Confusion Matrix:
[[28  9]
 [ 9 41]]


-->Classification Report:
              precision    recall  f1-score   support

           0       0.76      0.76      0.76        37
           1       0.82      0.82      0.82        50

    accuracy                           0.79        87
   macro avg       0.79      0.79      0.79        87
weighted avg       0.79      0.79      0.79        87



-->Accuracy Score: 0.7931034482758621


Random Forest Classsifier's AUC Score is 0.7884
```

Figure 3: Random Forest Classifier (RFC) report of the feature model.



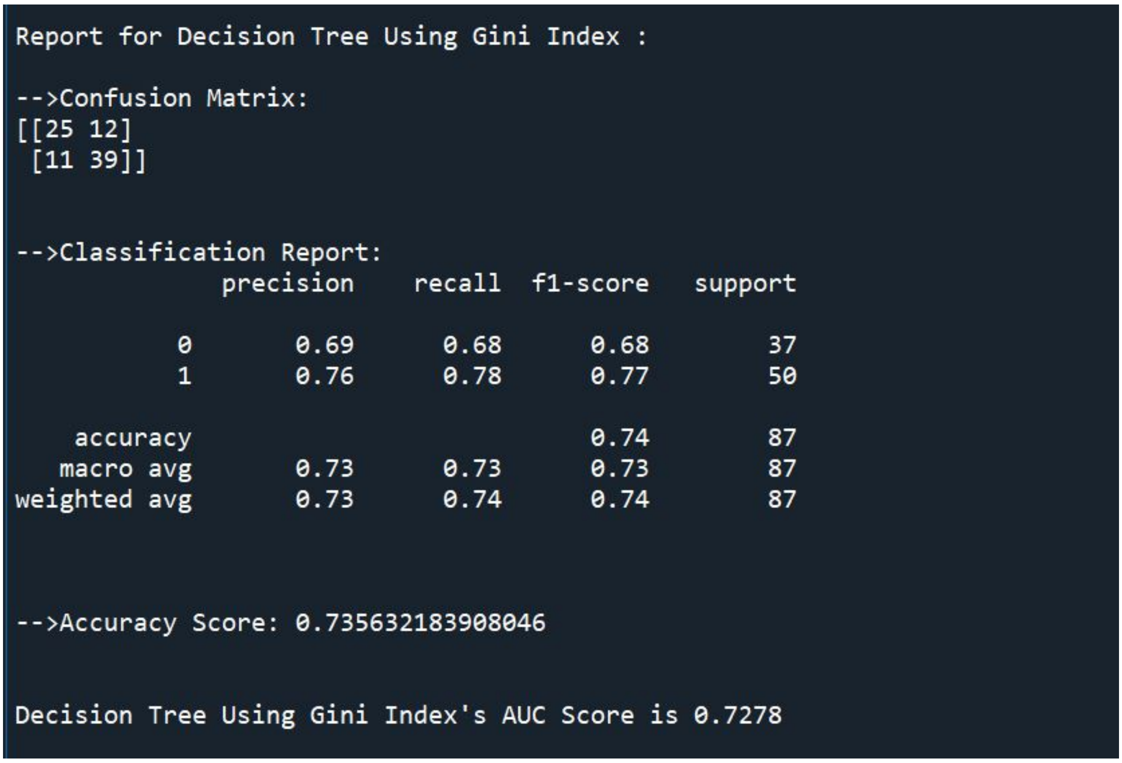
```
Report for K Nearest Neighbour Classifier :

-->Confusion Matrix:
[[21 16]
 [15 35]]


-->Classification Report:
              precision    recall  f1-score   support

           0       0.58      0.57      0.58        37
           1       0.69      0.70      0.69        50

    accuracy                           0.64        87
   macro avg       0.63      0.63      0.63        87
weighted avg       0.64      0.64      0.64        87



-->Accuracy Score: 0.6436781609195402


K Nearest Neighbour Classifier's AUC Score is 0.6338
```

Figure 4: K Nearest Neighbour Classifier (K-NN) report of the feature model.



```
Report for Decision Tree Using Entropy :

-->Confusion Matrix:
[[21 16]
 [10 40]]


-->Classification Report:
              precision    recall  f1-score   support

           0       0.68      0.57      0.62        37
           1       0.71      0.80      0.75        50

    accuracy                           0.70        87
   macro avg       0.70      0.68      0.69        87
weighted avg       0.70      0.70      0.70        87



-->Accuracy Score: 0.7011494252873564


Decision Tree Using Entropy's AUC Score is 0.6838
```

Figure 5: Decision Tree using Entropy report of the feature model.



```
Report for Decision Tree Using Gini Index :

-->Confusion Matrix:
[[25 12]
 [11 39]]


-->Classification Report:
              precision    recall  f1-score   support

           0       0.69      0.68      0.68        37
           1       0.76      0.78      0.77        50

    accuracy                           0.74        87
   macro avg       0.73      0.73      0.73        87
weighted avg       0.73      0.74      0.74        87



-->Accuracy Score: 0.735632183908046


Decision Tree Using Gini Index's AUC Score is 0.7278
```

Figure 6: Decision Tree using GINI index report of the feature model.

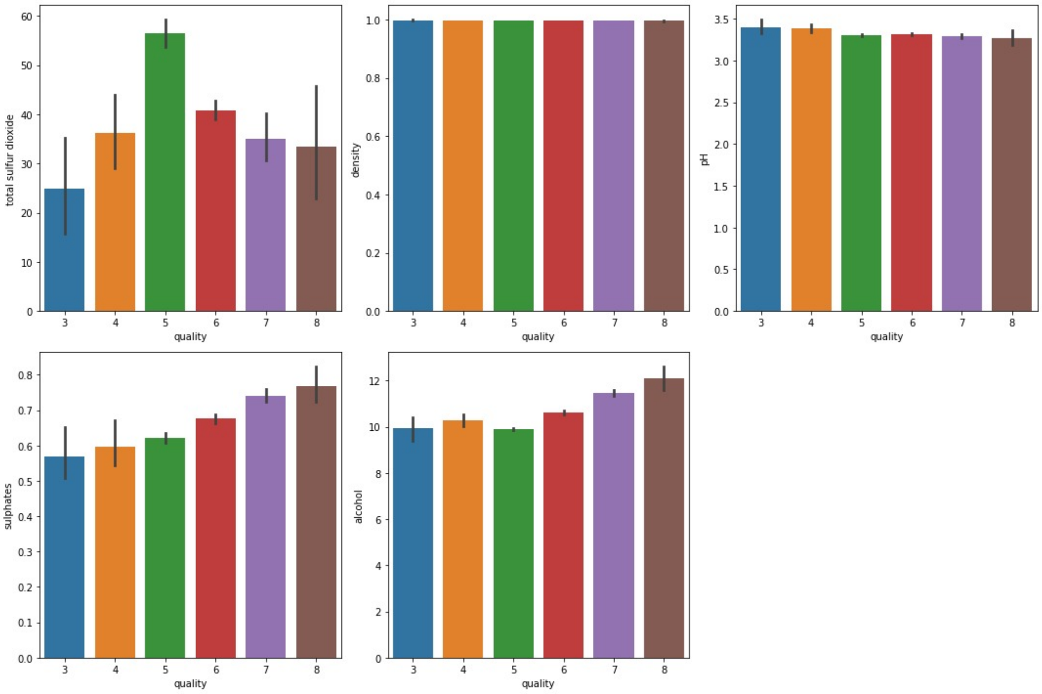


Figure 7: Quality Testing report of the feature model corresponding to Red Wine.

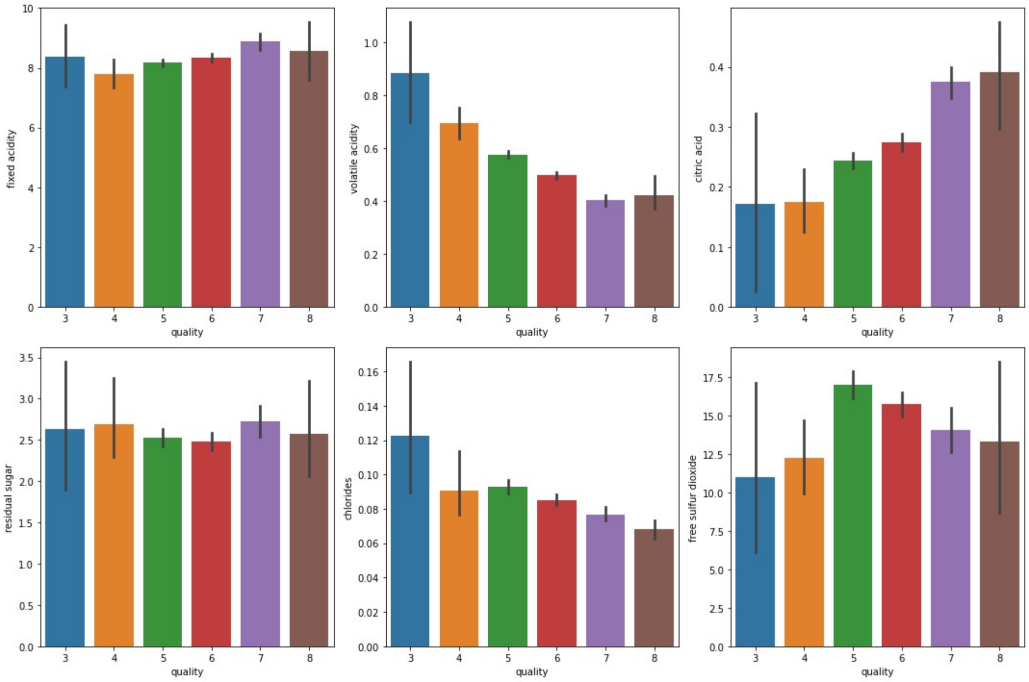


Figure 8: Quality Testing report of the feature model corresponding to White Wine.

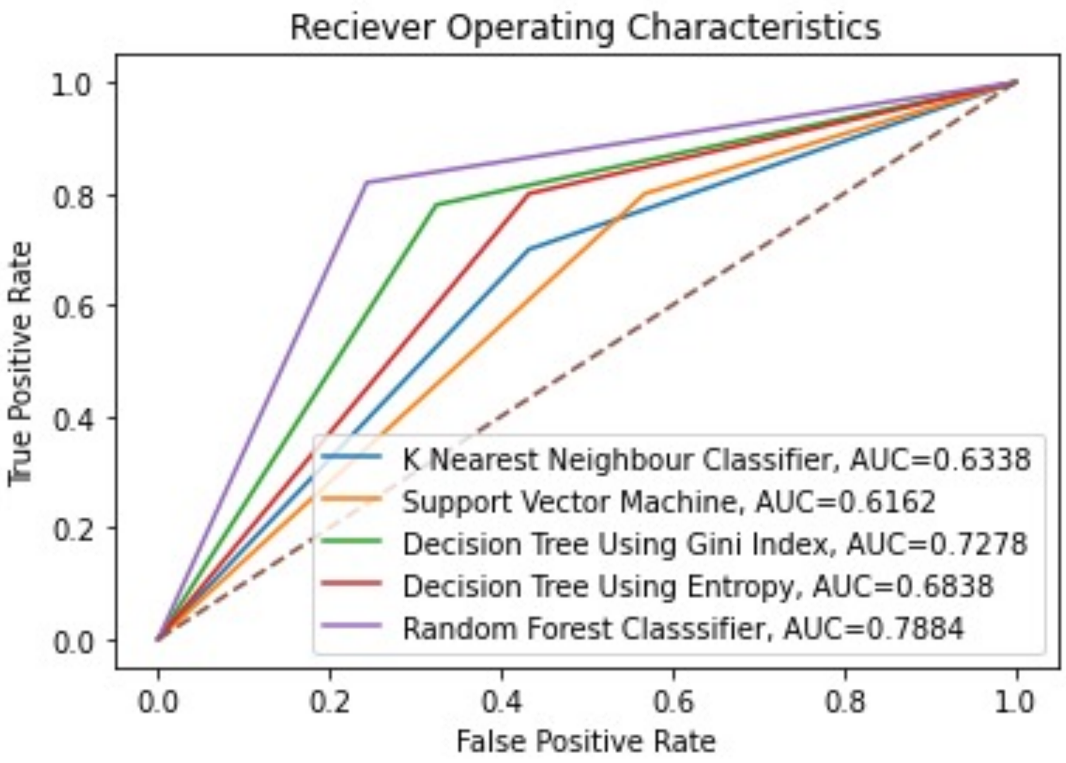


Figure 9: Receiver Operating Characteristics report of the feature model.

Conclusion and Future Scope

- The Random Forest Classifier has been introduced in this Wine Quality Prediction and this has given the highest AUC and accuracy score respectively as compared to the other classifiers since it uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.

- In the future, broad data set may be used for experiments and other machine learning techniques may be explored for the prediction of wine quality and to improve the accuracy of the classifier, it is clear that the algorithm or the data must be adjusted. We recommend feature engineering, using potential relationships between wine quality, or applying the boosting algorithm on the more accurate method.

References

[1] Dahal, K., Dahal, J., Banjade, H., Gaire, S., 2021. Prediction of Wine Quality Using Machine Learning Algorithms. Open J. Stat. 11, 278–289.

[2] Er, Y., Atasoy, A., 2016. The Classification of White Wine and Red Wine According to Their Physicochemical Qualities. Int. J. Intell. Syst. Appl. Eng. 4, 23–26.

[3] Gupta, Y., 2018. Selection of important features and predicting wine quality using machine learning techniques. Procedia Comput. Sci. 125, 305–312.

[4] Kumar, S., Agrawal, K., Mandan, N., 2020. Red Wine Quality Prediction Using Machine Learning Techniques, in: 2020 International Conference on Computer Communication and Informatics (ICCCI). Presented at the 2020 International Conference on Computer Communication and Informatics (ICCCI), IEEE, Coimbatore, India, pp. 1–6.

[5] P. Appalasamy, N.D. Rizal, F. Johari, A.F. Mansor, A. Mustapha, 2012. Classification-based Data Mining Approach for Quality Control in Wine Production [WWW Document].

[6] Lee, S., Park, J., Kang, K., 2015. Assessing wine quality using a decision tree, in: 2015 IEEE International Symposium on Systems Engineering (ISSE). Presented at the 2015 IEEE International Symposium on Systems Engineering (ISSE), IEEE, Rome, Italy, pp. 176–178.