Project 8: Decision Tree and Neural Networks on Red Wine Quality Dataset

Hari Chandana Kotnani and Sriya Gorrepati

Introduction:

Our project aims to analyze the factors that influence the quality of red wine, specifically focusing on the red variant of the Portuguese "Vinho Verde" wine. We recognize that the vinification process, location of grape cultivation, and grape types used in wine production are key factors in determining the quality of wine. To ensure quality control and provide consumers with accurate information, proper wine traceability is crucial. With this in mind, our project provides winemakers with essential information on the features that are crucial in making high-quality wine. Our analysis aims to empower winemakers to focus on these essential features to achieve the best possible quality of wine.

Slides: Slides Link Github: Github Link

Dataset:

The red wine quality dataset, chosen from Kaggle, comprises 1599 samples of red wine. Each sample is described by 12 attributes, including Fixed acidity, Volatile acidity, Citric acid, Residual sugar, Chlorides, Free sulfur dioxide, Total sulfur dioxide, Density, pH, Sulfates, Alcohol, and Quality. Important attributes, such as Alcohol, Volatile acidity, Sulfates, Citric acid, and Total sulfur dioxide, have been identified as playing a vital role in wine making. The dataset is suitable for classification and regression tasks related to wine production, marketing, and quality control, and can provide valuable insights into the factors that influence wine quality. The large number of samples and attributes in the dataset make it ideal for decision tree and neural network projects. Decision trees can be used to identify the most important attributes affecting wine quality, while the dataset can be used to train neural networks to predict wine quality based on various physical and chemical properties.

Dataset Link: Dataset Link

Data Preprocessing:

For decision tree classification, In the binary classification approach, we converted the 'quality' attribute and binarized the data by converting values below 5 to '0', considering values below 5 as 'bad' and values above 5 to '1', considering values above 5 as 'good'. For our neural network model, we applied the StandardScaler preprocessing step to the data by fitting it to the training data and then transforming both the training and test data.

Analysis Technique:

Our project involves exploratory analysis using decision trees and neural networks. For the decision tree analysis, we approached the classification problem in two ways: multi-class and binary classification. In the multi-class approach, we treated each quality value from 3 to 8 as a separate class, while in binary classification, we treated quality values above 5 as one class and below or equal to 5 as another class. We analyzed the correlation between each pair of features in the dataset and selected the top five features with the highest correlation with the target variable taking those five as feature columns for the input data and target variable 'quality' represents the output data. We utilized a decision tree classifier to interpret the data and predict the samples accurately. This approach helped us handle outliers by putting them aside in

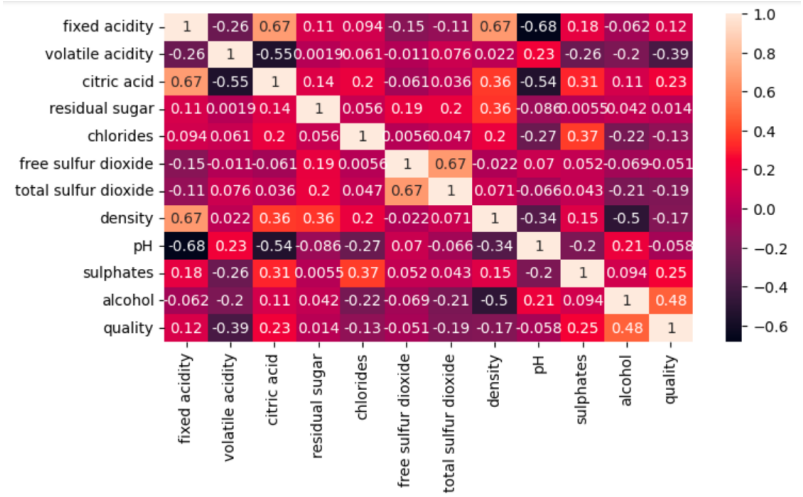
separate leaf nodes, which allowed us to predict the quality accurately. We also used GridSearchCV to find the best parameters for the model and visualized a tree, including other parameters like criterion, max_leaf_nodes, and min_samples_leaf.

The second technique we employed is neural networks. They are considered powerful machine learning models that can effectively model complex relationships between inputs and outputs. In the case of wine quality prediction, neural networks can learn from extensive datasets of wine characteristics such as acidity, pH, alcohol content, and other factors that impact wine quality. To achieve the best possible accuracy, we utilized MLP classifiers and implemented a grid search by looping through various hidden layer sizes to select the best three models. We evaluated the performance of the three best models selected by the grid search.

Results:

Decision tree:

In the decision tree, we use the correlation matrix to identify the most relevant features, as shown in graph1. Based on this analysis, we determined that 'alcohol', 'volatile acidity', 'sulphates', 'citric acid', and 'total sulfur dioxide' are the best features for our model.



Graph.1 correlation matrix

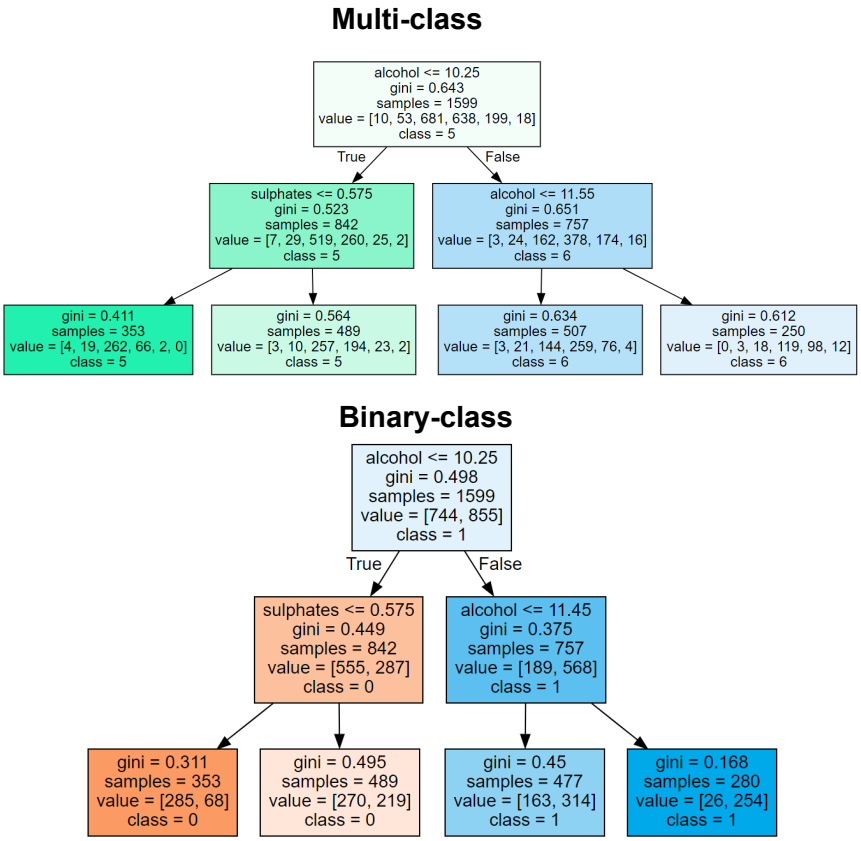
For visualizing decision trees, we have decided to set the maximum depth of the trees to 2 and 5 for both approaches, which allows us to compare and contrast the differences in the structure and complexity of the resulting decision trees at different levels of depth.

To calculate feature importance we used the Gini importance, which measures the total reduction of impurity that a feature contributes to all the splits in a decision tree.

Max Depth 2: The feature importance (using the Gini importance method) results while using the decision tree for max_depth value 2 is,

```
alcohol: 0.854680408049223
volatile acidity: 0.0
sulphates: 0.14531959195077704
citric acid: 0.0
total sulfur dioxide: 0.0
```

As per the results, alcohol has the highest importance score of 0.85, followed by sulphates with a score of 0.15. In contrast, volatile acidity, citric acid, and total sulfur dioxide have zero importance scores, indicating that they do not contribute to the decision-making process of the tree.The decision tree starts by splitting the data based on the attribute with the highest importance score, which is alcohol in this case. Subsequently, the remaining attributes are divided into two groups based on their condition values relative to alcohol. The algorithm aims to minimize impurities by selecting attributes with lower Gini values, which determine the splitting for the subsequent levels of the tree. As a result, this addresses the question of which attributes the algorithm chooses to split on.



Graph2 decision tree of max_depth 2

Here, we can see that the sulphate gini is having the least value 0.411 compared to all the other values with 353 samples. Alcohol has the highest no.of samples with 757 but we have to note that having the highest number of samples can lead to overfitting.

Both multi-class and binary classifiers were trained using the same hyperparameters and resulted in the same trees and feature importances. The only difference was the class labels assigned to the outputs, where the first classifier assigned the classes ['3', '4', '5', '6', '7', '8'] and the second classifier assigned the classes ['0', '1']."

Also, For a maximum depth of 2, based on the result tree we can see that at the first level of the tree, the algorithm chose to split on alcohol with a threshold value of 10.3. At the second level of the tree, the algorithm chose to split on sulphates and again with higher alcohol threshold. Therefore, we can conclude that not all nodes split on the same attribute at a given tree level.

Approach 1: After training the decision tree with multi-class approach, this is the classification chart that we are getting,

Accuracy: 0.59375

	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	13
5	0.63	0.78	0.69	140
6	0.55	0.60	0.58	134
7	0.00	0.00	0.00	30
8	0.00	0.00	0.00	2
accuracy			0.59	320
macro avg	0.20	0.23	0.21	320
weighted avg	0.51	0.59	0.55	320

We can see that in the above metrics, the values are calculated only for 5 and 6 classes since all the other classes' values are 0. The values generated in both the classes are similar to one another.

Approach 2: We implemented this to compare both binary class and multi-class performancel and see which one could give a better output. After training the decision tree with binary approach, this is the accuracy that we are getting,

Accuracy : 0.70625

Based on the given results, we can conclude that the binary classification approach performed better than the multi-class approach. The accuracy of the binary classification (70.625%) is higher than the accuracy of the multi-class classification (58.75%). This suggests data preprocessing such as a binary classification task (above 5 vs below or equal to 5) may have been a more effective approach for this particular dataset.

Max Depth 5: The feature importance (using the Gini importance method) results while using the decision tree for max_depth value 5 is,

```
alcohol: 0.4701451258266833
volatile acidity: 0.10130562881723514
sulphates: 0.20798219808634727
citric acid: 0.018271224911649526
total sulfur dioxide: 0.2022958223580848
```

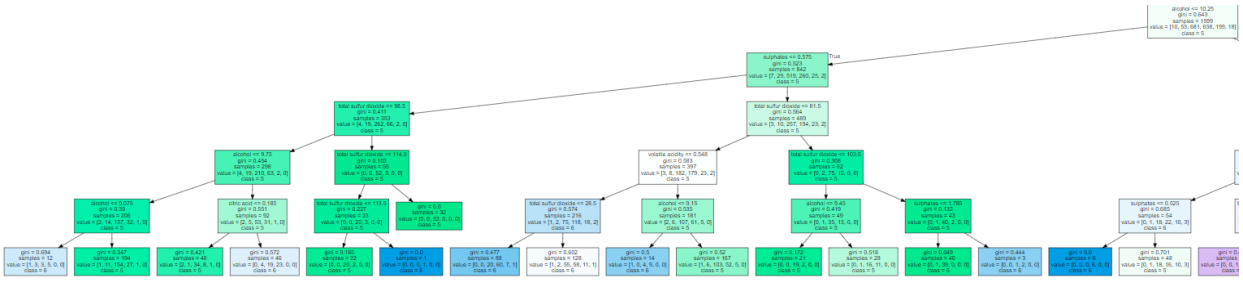
Multi-class

```
alcohol: 0.5091813094225128
volatile acidity: 0.09609707716101944
sulphates: 0.22966563969990947
citric acid: 0.02057261994318562
total sulfur dioxide: 0.14448335377337274
```

Binary-class

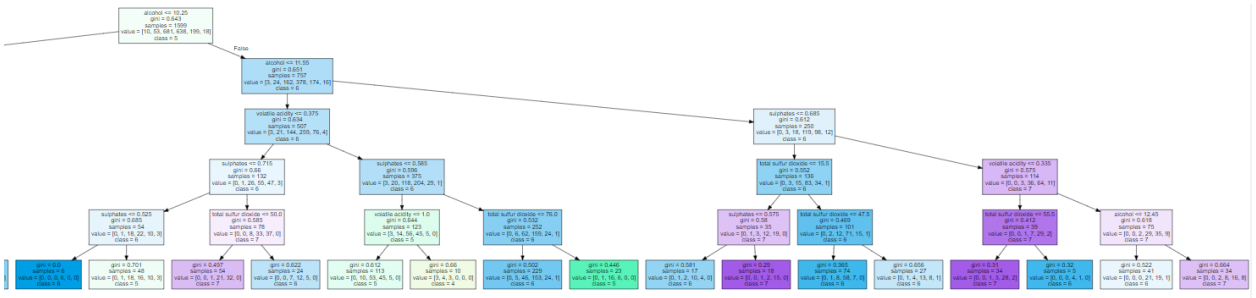
The values associated with each feature in the two trees are different, but it is important to note that the priorities of the features are the same. This means that the relative importance of each feature in determining the outcome of the decision tree is the same in both cases.As a result, the two decision trees have the same structure and the same decision-making process, except for the fact that the classes are different. Therefore, the graph representing the decision trees would also be the same except for the classes.

Alcohol has the highest value amongst other features same as in the max_depth 2. Though the value of alcohol is slightly less than the previous value in max_depth 2, remaining features such as volatile acidity, citric acid and total sulfur dioxide are also having a significant role making a place for themselves. The algorithm has selected "alcohol" as the attribute with the highest value and splits, followed by "sulphates" and "volatile acidity" then "total sulphur dioxide" and "citric acid".
Here, since max depth 5 graph is huge we had to divide it into 2 parts



Graph 4 Alcohol condition is true

The above graph shows how the tree is split for the alcohol condition 'true' where it is <=10.25.



Graph 5 Alcohol condition is false

Note : As the graphs for both classes with a maximum depth of 5 were large, we added them in PNG format to our GitHub report as graphical representations.

For max depth 5, we can see that at the first level of the tree, the algorithm chose to split on alcohol with a threshold value of 10.3. At the second level of the tree, the algorithm chose to split on sulphates and again alcohol. At the third level, the algorithm chose to split on feature

total sulphur dioxide and volatile acidity and then the next level again sulphates and alcohol. Therefore, we can again conclude that not all nodes split on the same attribute at a given tree level.

Approach 1: After training the decision tree with multi-class approach, this is the classification chart that we are getting,

Accuracy: 0.5875

	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	13
5	0.62	0.74	0.68	140
6	0.60	0.54	0.57	134
7	0.41	0.40	0.41	30
8	0.00	0.00	0.00	2
accuracy			0.59	320
macro avg	0.27	0.28	0.28	320
weighted avg	0.56	0.59	0.57	320

Approach 2: After training the decision tree with binary approach, this is the accuracy that we are getting, Accuracy: 0.746875

Similar to max depth 2 performance, the binary classification approach performed better than the multi-class approach. Multi-class accuracy is 0.5875. On the other hand, the reported binary-class accuracy is 0.746875.Based on the results, it can be concluded that the binary classification approach outperformed the multi-class classification approach for this particular dataset. The binary classification accuracy is significantly higher than the multi-class accuracy.

Hyperparameters:

As the accuracy was low for both 2 and 5 max depths in the multi-class approach, we attempted to improve the performance by adding more hyperparameters. We used grid search to test different values for various parameters and selected the best ones. Then, we created a graph using these optimized parameters and evaluated the performance to compare it with the previous results.

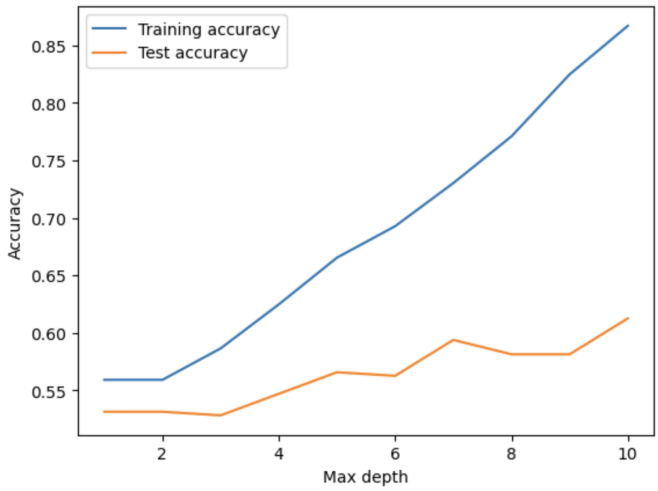
```
Best parameters:  {'criterion': 'entropy', 'max_depth': 5, 'max_leaf_nodes': None, 'min_samples_leaf': 5}
```

Accuracy: 0.59375

	precision	recall	f1-score	support
3	0.00	0.00	0.00	1
4	0.00	0.00	0.00	13
5	0.66	0.68	0.67	140
6	0.57	0.59	0.58	134
7	0.41	0.53	0.46	30
8	0.00	0.00	0.00	2
accuracy			0.59	320
macro avg	0.27	0.30	0.29	320
weighted avg	0.57	0.59	0.58	320

It seems that adding more hyperparameters through grid search did not significantly improve the accuracy of the model. The accuracy remained around 0.59 for all three cases: max depth 2, max depth 5, and the best parameters. (Graph uploaded in Graphs of Report folder in git)

Effect of depth on bias and variance: As we can see from the plot, the performance of a model was evaluated by varying the max_depth from 1 to 10, and recording the training and test accuracies. The results showed that as the max_depth increased, the training accuracy improved, indicating a reduction in bias. However, the test accuracy initially improved but then plateaued or even decreased, indicating an increase in variance. The optimal balance between bias and variance was achieved at max_depth = 7, which resulted in a good test accuracy. Further increasing the max_depth resulted in overfitting and a decrease in test accuracy. Therefore, finding the right balance between bias and variance is crucial for achieving good model performance..



When examining the correlations between edge weights and features using a decision tree and gini index, we observed that 'alcohol' had a large positive weight, indicating that higher alcohol content in red wine is positively associated with higher quality ratings. Conversely, we also noted that 'volatile acidity' had a large negative weight, suggesting that lower volatile acidity is negatively correlated with higher quality ratings.

Neural Networks:

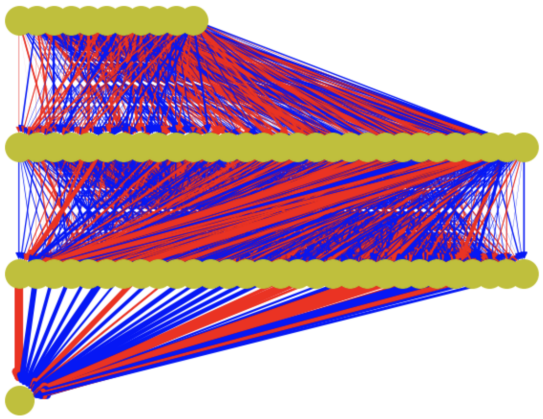
We decided to use the same binary-class data that performed well with decision tree for neural networks. The dataset divides quality into two classes, and we chose to use all features without selecting any best features for input. To visualize the trained MLP model, we used the networkx package, which generates a graph showing the structure and connections of the neural network model. This graph can be useful in understanding how the model is making predictions. After standardizing the data, we trained a multi-layer perceptron (MLP) classifier using grid search CV, where we randomly tried different hidden layer sizes with max_iter as 10000, and selected the best model based on its performance.

```
Configuration  1 :
Parameters: {'hidden_layer_sizes': (30, 30)}

Configuration  2 :
Parameters: {'hidden_layer_sizes': (30, 30, 15)}

Configuration  3 :
Parameters: {'hidden_layer_sizes': (15, 15, 15)}

MLPClassifier(hidden_layer_sizes=(30, 30), max_iter=10000)
```

In Neural Network, the first hidden layer takes the input and applies weights and biases to produce an output, which is then passed on to the second hidden layer. This process is repeated for the third hidden layer before the output layer produces the final result.

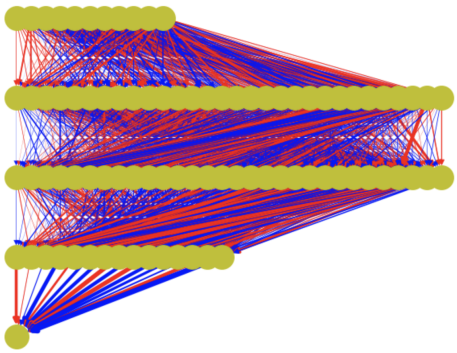
Architecture 1: with hidden layer sizes as (30, 30), 2 hidden layers, each with 30 neurons.

Results for parameters (30,30),



	precision	recall	f1-score	support
0	0.68	0.71	0.69	184
1	0.74	0.71	0.73	216
accuracy			0.71	400
macro avg	0.71	0.71	0.71	400
weighted avg	0.71	0.71	0.71	400

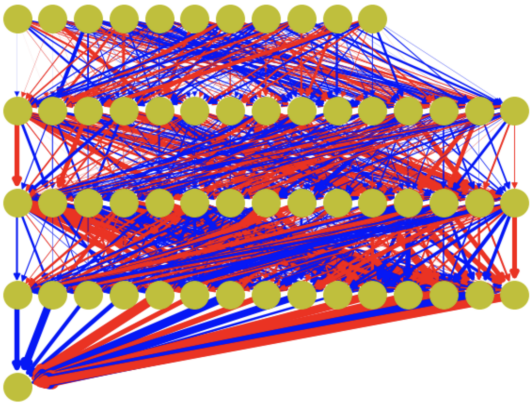
Architecture 2 : with hidden layer sizes as (30, 30, 15), 3 hidden layers, first and second containing 30 neurons and 15 neurons in third layer.



Results for parameters (30,30,15),

	precision	recall	f1-score	support
0	0.75	0.65	0.69	184
1	0.73	0.81	0.77	216
accuracy			0.74	400
macro avg	0.74	0.73	0.73	400
weighted avg	0.74	0.74	0.74	400

Architecture 3 : with hidden layer sizes as (15, 15, 15) ,3 hidden layers, each containing 15 neurons.



Results for parameters (15,15,15)

	precision	recall	f1-score	support
0	0.74	0.72	0.73	184
1	0.76	0.78	0.77	216
accuracy			0.75	400
macro avg	0.75	0.75	0.75	400
weighted avg	0.75	0.75	0.75	400

Parameter (15,15,15) is having highest accuracy and f1 score.
Out of the three classification reports provided, the second one (30,30,15) displays the highest values for precision, recall, and f1-score for both classes, along with the highest support value for class 1. This indicates that it is the most successful classification report among the three. Also, there is no direct correlation between the edge weights of a neural network and decision trees for this dataset.

After comparing the performance of neural networks and decision trees on our dataset, we found that the best accuracy achieved by neural networks was 0.75 using parameters (15,15,15), while the decision tree(depth =5) trained with a binary approach had an accuracy of 0.746875. These results suggest that neural networks and DT have similar performance on this dataset. As an alternative approach, it may be worth exploring feature selection techniques, like what we did in decision trees, that might improve the accuracy of the neural networks model.