Durham University

# Wine Quality: The Good, the Bad, and the Average

Tree-based Machine Learning Models to Categorize Wine Qualities through Physicochemical Properties

Master of Data Science

Academic year 2023/2024

---

**Dataset Topic:**

Wine Quality – model wine quality based on physicochemical tests using both the red and white wine data sets.

**Report Title:**

Wine Quality: The Good, the Bad, and the Average – Tree-based Machine Learning Models to Categorize Wine Qualities through Physicochemical Properties

**Anonymous Marking Code:** Z0194160

**Postgraduate Programme:** MDS – Master of Data Science

**Module:** Machine Learning (MATH42815)

**Word Count (3000 words - maximum):** 2997 words

---

Department of Mathematical Sciences

Durham University

Durham, United Kingdom

February 12, 2024

# 1. Introduction

## 1.1 Problem Description

With a rich history spanning nearly 6,000 years, wine has played a significant role in human culture, with early wine production evidence in Geogia (Soleas et al., 1997). From ancient civilization to modern times, wine transcended cultural and geographical boundaries of Europe into a diverse array of consumers around the globe. Nevertheless, majority of the wine is still being produced in Europe with an average production share of 65.5% from 1994 to 2022 (FAOSTAT, 2023). To safeguard regional wine authenticity and quality, the European Union (EU) has established the Wine Protected Designation of Origin (PDO) labeling system to ensure that wine adhere to its geographical origin standards (Candiago, 2022). Despite technological advancements, wine quality assessment still relies on oenologist expertise, which is time consuming and costly process (Cortez et al., 2009a). This has led to growing research interest in applying machine learning (ML) techniques to quality wine quality through physicochemical patterns (Bhardwaj et al., 2022; Cortez et al., 2009a). This report will explore tree-based ML approaches to model “Vinho Verde” wine quality through collected physicochemical characteristics, which was previously studied by Cortez and collaborators (2009a). Similar to their focus in supporting oenologist decision-making, this report aims to explore ML models capable of reducing oenologist workload by automatic screening of wine quality.

## 1.2 Wine Dataset Overview

This report utilizes a wine dataset containing a collection of red and white wines originated from the Northern part of Portugal in a region referred to as “Vinho Verde”, available at the University of California, Irvine (UCI) online Machine Learning Repository (Cortez et al., 2009b). This wine dataset consists of two comma-separated value (.csv) files containing 1,599 observations of red and 4,898 observations of white wine. Each observation contains 11 physicochemical attributes and it respected median wine grading score on the scale of 0 (awful) to 10 (excellent), which is based on three blind sensory tests by wine experts (Cortez et al., 2009a). As this analysis objective is to model the wine quality for both red and white wine, an additional variable of wine color is added as one of the variables. This is due to the inherent difference in sensory experience for both wine types, which is also noted by Cortez and colleagues (2009a). Several researchers also support this observation where the visual perception of color influences human perceived taste (Pangborn et al., 1963; Parr et al., 2003; Spence et al., 2010). Therefore, wine color is added as one of the feature variables in this analysis. A table that summarizes wine dataset variable details and their descriptions is provided in Table 1.

*Table 1: Wine Dataset Variables*

| Name | Units | Role | Type | Description |
|---|---|---|---|---|
| Fixed Acidity | $g/dm^3$ | Feature | Continuous | Measure of soluble tartaric acid in the solution |
| Volatile Acidity | $g/dm^3$ | Feature | Continuous | Measure of soluble acetic acid in the solution |
| Citric Acid | $g/dm^3$ | Feature | Continuous | Measure of citric acid in the solution, contributing to wine freshness taste. |
| Residual Sugar | $g/dm^3$ | Feature | Continuous | Measure of leftover sugar from the fermentation process, contributing to wine sweet taste. |
| Chlorides | $g/dm^3$ | Feature | Continuous | Measure of salt in the solution |
| Free Sulfur Dioxide | $mg/dm^3$ | Feature | Continuous | Amount of unbound of sulfur dioxide |
| Total Sulfur dioxide | $mg/dm^3$ | Feature | Continuous | Total amount of bound and unbound sulfur dioxide |
| Density | $g/cm^3$ | Feature | Continuous | Measurement to evaluate the fermentation process |
| pH | *NA* | Feature | Continuous | Measure alcohol solution acidity or basicity |
| Sulphates | $g/dm^3$ | Feature | Continuous | Measure of potassium sulphate |
| Alcohol | Volume % | Feature | Continuous | Percent of the wine alcohol content |
| Color | *NA* | Feature | Categorical | Red or white wine |
| Wine Quality | *NA* | Target | Integer | Oenologist score rating between 0 and 10 |

*(Note:* Apated from *Modeling wine preferences by data mining from physicochemical properties,* by Cortez et al., 2009a *and Compendium of international methods of wine and must analysis, by OIV., 2022)*

### 1.3 Proposed Modelling Approach

This report will be exploring tree-based classification methods, specifically decision tree and random forest ML algorithms, for this wine grading application. The software utilized for this analysis is R, available on the NVIDIA CUDA Centre (NCC), to statistically explore the dataset and build ML models. During the exploratory data analysis, it revealed that the wine quality data contains only 3 to 8 rated scores for red and 3 to 9 scores for white wine, respectively. Given that the oenologist's potential rating for wine quality is an integer between 0 and 10, regression or classification ML model based on the available data would fail to account for observations out of the collected dataset range. Therefore, under the assumption that this model will be deployed in the real-world, we will group the wine quality rating score into three categories: good wine (7 and above rating), average wine (5 and 6 rating), and bad wine (0 to 4 rating), to maintain grading function while accommodating for potential future rating scores.

# 2. Dataset Preparation and Exploratory Data Analysis

## 2.1 Dataset Preparation and Data Cleaning

With the two datasets loaded, we will verify the column data types are double precision for feature variables and integers for the response variable. Moreover, there are no missing values (*NA*) for either data frame. For data duplication, we found out that 240 out of 1,599 entries (15.01%) of red and 937 out of 4,898 entries (19.13%) of white wine contained duplicated values. Nevertheless, we will not be removing these duplicated values as wine being fermented in batches could lead to the same or similar level of physicochemical levels. As this model will be focusing on utilizing both wine datasets, we will combine these two data frames into one. To accommodate for the differences in taste perception between red and white wine, which has been noted by multiple research units (Cortez et al., 2009a; Pangborn et al., 1963; Parr et al., 2003; Spence et al., 2010), a column for color will be added for each category of wine either “red” or “white” as a factor <fct> variable. After binding, the column names will be modified to the same format by using “_” to replace “.” in the names. Additionally, we will create a column for the three categories of wine: 0-4 (bad), 5-6 (average), and 7-10 (good). With the data prepared, we will move to data exploration.

## 2.2 Exploratory Data Analysis

### 2.2.1 Data Distribution Exploration

In this section, we will explore the data to the overview of the dataset and its trends. First, we create a five-number summary of the wine data, Figure 1. From the figure, we found that the majority of the data is white wine, taking up 75.39% of the data set, compared to red wine at 24.61%, from the total of 6,497 observations. For the target variable, there are a considerable number of observations in the 5 and 6 quality range compared to other ranges. Histogram plot for the target variable as the recorded score quality (A.) and quality ranges (B.), Figure 2, revealed that the distribution of the score is of a normal distribution (roughly bell shape) for both wine colors with the majority of wine grades lies around the average grade (5 and 6 scores). When grouping the data into three ranges of quality, wine scores of 0 to 4 have the lowest representation in our wine quality data. This could potentially lead to lower ML model predictive accuracy for this bad wine category.

```
fixed_acidity    volatile_acidity  citric_acid     residual_sugar
Min.   : 3.800   Min.   :0.0800   Min.   :0.0000   Min.   : 0.600
1st Qu.: 6.400   1st Qu.:0.2300   1st Qu.:0.2500   1st Qu.: 1.800
Median : 7.000   Median :0.2900   Median :0.3100   Median : 3.000
Mean   : 7.215   Mean   :0.3397   Mean   :0.3186   Mean   : 5.443
3rd Qu.: 7.700   3rd Qu.:0.4000   3rd Qu.:0.3900   3rd Qu.: 8.100
Max.   :15.900   Max.   :1.5800   Max.   :1.6600   Max.   :65.800

  chlorides       free_sulfur_dioxide total_sulfur_dioxide    density
Min.   :0.00900   Min.   :  1.00      Min.   :  6.0        Min.   :0.9871
1st Qu.:0.03800   1st Qu.: 17.00      1st Qu.: 77.0        1st Qu.:0.9923
Median :0.04700   Median : 29.00      Median :118.0        Median :0.9949
Mean   :0.05603   Mean   : 30.53      Mean   :115.7        Mean   :0.9947
3rd Qu.:0.06500   3rd Qu.: 41.00      3rd Qu.:156.0        3rd Qu.:0.9970
Max.   :0.61100   Max.   :289.00      Max.   :440.0        Max.   :1.0390

      pH          sulphates         alcohol      quality    color
Min.   :2.720   Min.   :0.2200   Min.   : 8.00   3:  30   red  :1599
1st Qu.:3.110   1st Qu.:0.4300   1st Qu.: 9.50   4: 216   white:4898
Median :3.210   Median :0.5100   Median :10.30   5:2138
Mean   :3.219   Mean   :0.5313   Mean   :10.49   6:2836
3rd Qu.:3.320   3rd Qu.:0.6000   3rd Qu.:11.30   7:1079
Max.   :4.010   Max.   :2.0000   Max.   :14.90   8: 193
                                                 9:   5

quality_range
0-4 : 246
5-6 :4974
7-10:1277
```

*Figure 1: Wine Data Summary*

For feature variables, the five-number summary alongside histogram plots (Figure 3) revealed that chlorides, free sulfur dioxide, residual sugar, and sulphates are positively skewed. Moreover, the figures indicated that there are outliers present within the data, as some observations deviate from the distribution. Nevertheless, we will not be removed from inspection in the data set these physiochemical observations are possible due to the wine fermentation processing process, affecting the taste and, subsequently, the graded wine quality.

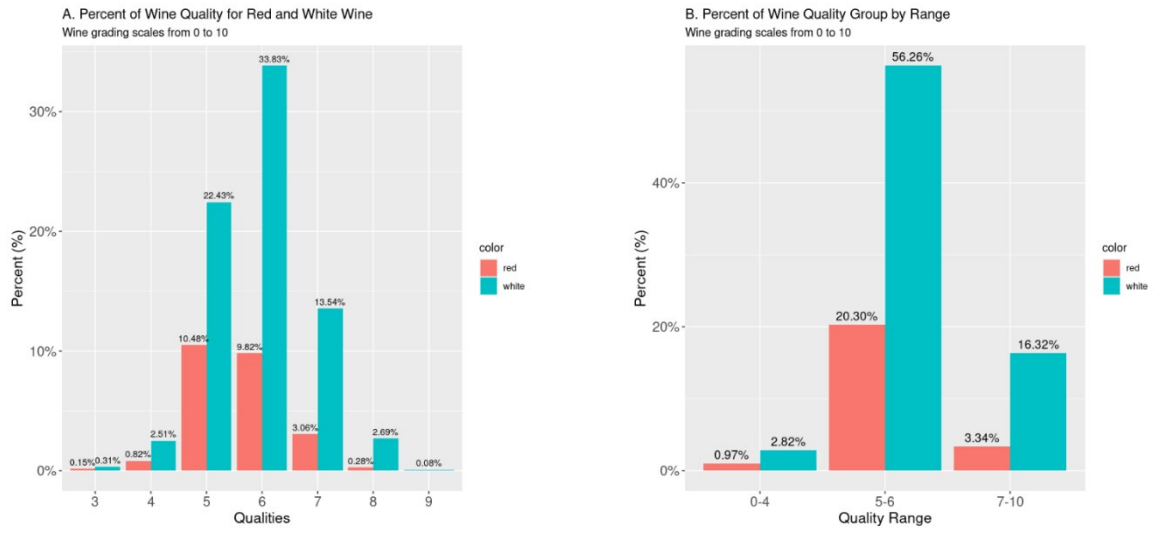



*Figure 2: Histogram of Wine Dataset (A.) figure showing the percentage of wine quality for red and white wines; (B.) figure showing the percentage of wine quality when grouped by quality range.*

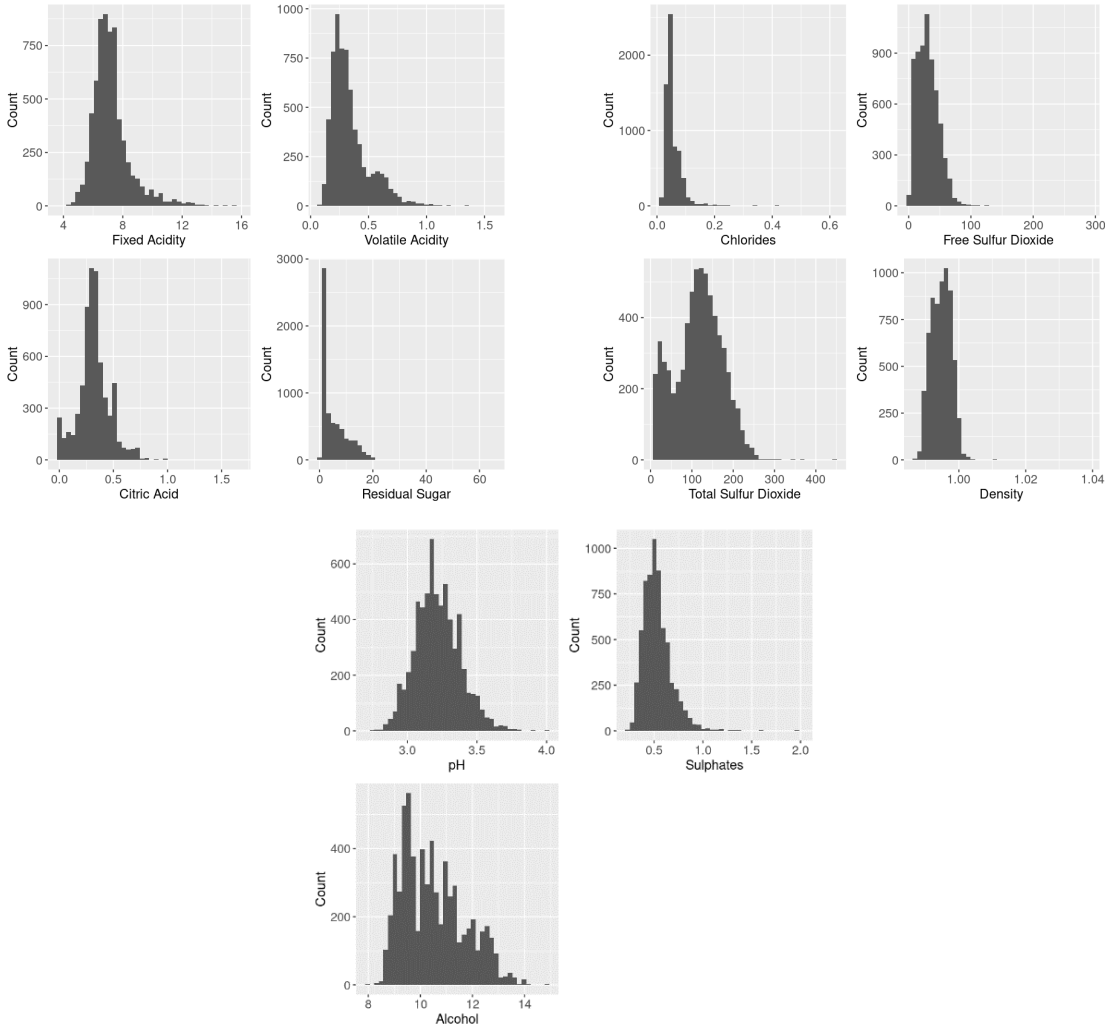



*Figure 3: Histogram of Wine Feature Variables (bins = 40)*

### 2.2.2 Response and Predictor Variables Relationship Analysis

To explore the relationship between the target and feature variables, we plot the boxplot of each quality range for all the feature variables, in Figure 4. We observed a positive relationship between wine's quality range and alcohol content (most prominent), citric acid, and free sulfur dioxide. As for the negative relationship, we identify the negative relationship between the assigned quality range and volatile acidity and density. For other predictor variables, there are no visible relationship with the quality range.

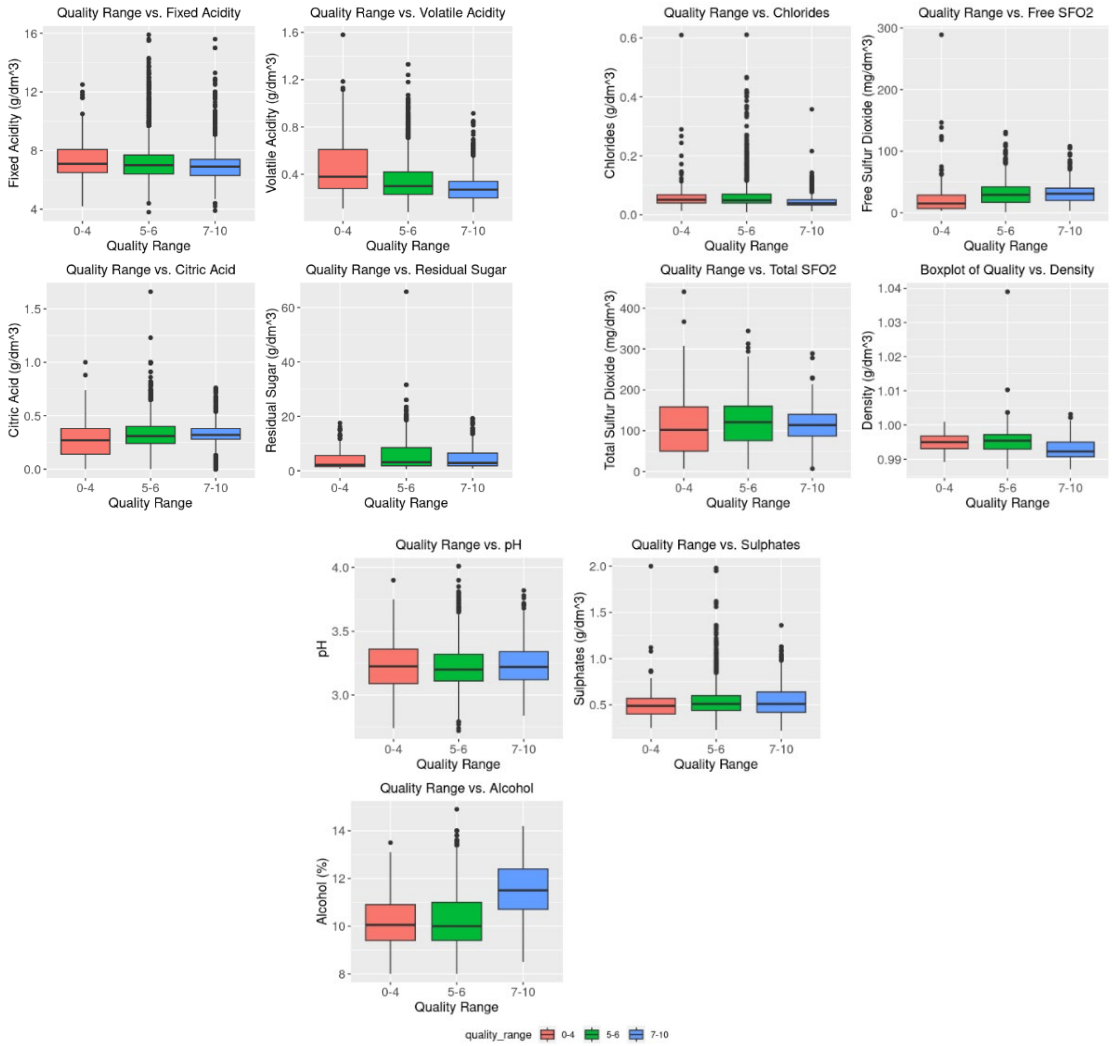



*Figure 4: Boxplots of Feature Variables in Defined Quality Range: 0-4 (red), 5-6 (green), and 7-10 (blue).*

To confirm this observation, a correlation matrix among feature and target variables for the wine dataset is plotted, in Figure 5. The correlation matrix provides the Pearson correlation coefficient between the variables. This correlation coefficient measures the strength of the linear relationship between variables, having a value between -1 (negative) and 1 (positive), with 0 as no correlation (Diez et al., 2019, pp. 310-311). From the figure, it is evident that alcohol has a moderate positive correlation with quality, others have relatively low positive correlation. Density, volatile acidity, and chloride have a negative correlation with the quality. A matrix of scatter, plotted with the “ggpairs” function, is provided in appendix a.

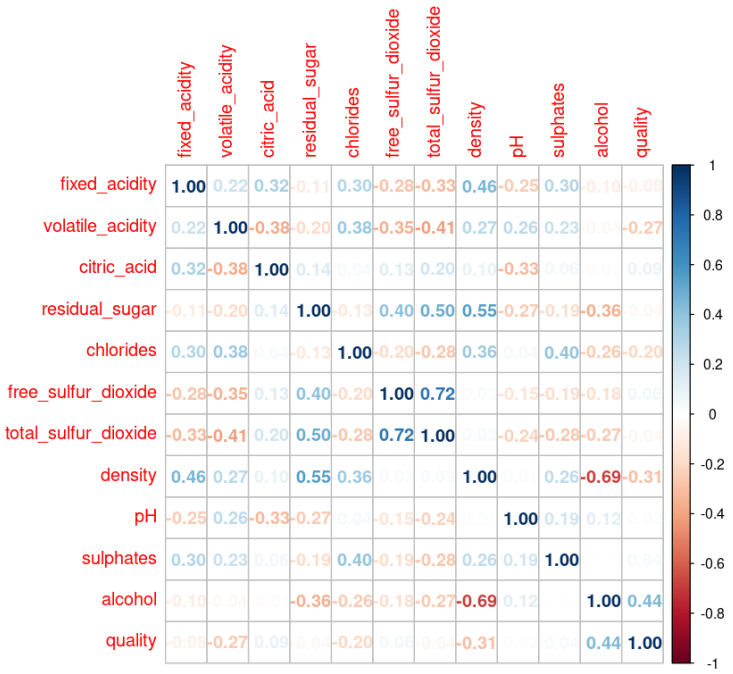



*Figure 5: Wine Data Correlation Matrix*

From this analysis, we hypothesized that alcohol, density, volatile acidity, and chlorides are potential features that will be considered as the branching point in the tree-based classification models. After exploring the data, we will move to supervised tree-based modeling.

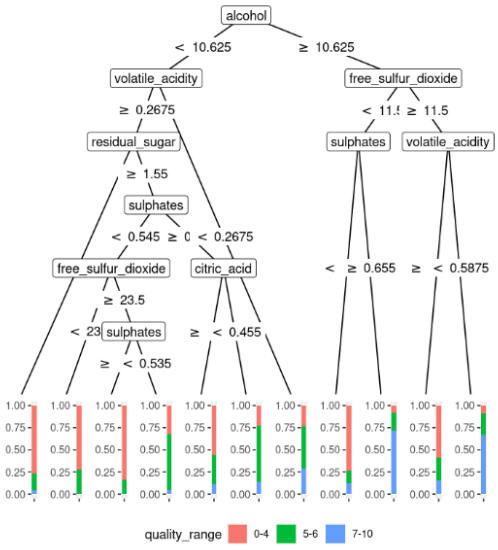
## 3. Supervised Machine Learning Modelling

### 3.1 Pre-modelling Process

The wine dataset is divided into training (70%) and testing (30%) to build models and evaluate the model, respectively. The “createDataPartition” is used to maintain the distribution between red and white wines. Additionally, as our wine data set is organized into orders of red and white wines, we will shuffle the dataset to mitigate potential bias during the test and train data splitting. More importantly, it is crucial to address the imbalance among the quality ranges to ensure that the model captures the distinct features of each class. According to our data, the predominant category is the average (76.56%), which is the score of 5 and 6, followed by good (19.65%) and bad (3.79%). As we are limited to the NCC environment, we will be using the upsampling method instead of the downsampling method, from the “caret’ library. This is to prevent downsampling information loss, as the training dataset will be reduced to the same proportion of the lowest category. The Upsampling function samples the minority categories to match the majority category number of samples. For reproducibility and consistency, all analyses in this report will use “set.seed(42)” to ensure reproducible results.

### 3.2 Classification and Regression Tree (CART)

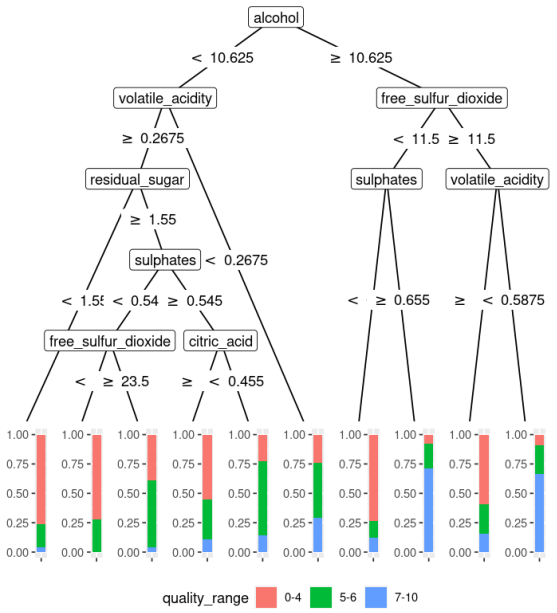
Tree-based models, also known as decision trees, are ML algorithms that recursively split the predictor space into small regions based on splitting points. These splitting points, referred to as decision nodes, are chosen to minimize the residual sum square (RSS), which is the sum of square differences between actual data and predicted values, for regression or to minimize misclassification of split regions for classification models (James et al., 2021). The algorithm will stop after a specific criterion is met. Given our application of categorizing wine, we will focus on classification tree-based models. In classification trees, a measurement referred to as the "Gini index", which has values between 0 (pure) and 1 (impure), determines the tree splitting points. It is a measure of category misclassification when a certain splitting point is selected (ibid). With the "rpart" function, a decision tree model is generated from the upsampling data (Figure 6, left). As the decision tree is prone to overfit the training data, we will print out the complexity parameters (CP) table during each tree-splitting stage (Figure 6, right).




```
Classification tree:
rpart(formula = quality_range ~ ., data = train_wine_up, method = "class")

Variables actually used in tree construction:
[1] alcohol             citric_acid         free_sulfur_dioxide
[4] residual_sugar      sulphates           volatile_acidity

Root node error: 6964/10446 = 0.66667

n= 10446

        CP nsplit rel error  xerror      xstd
1 0.228748      0   1.00000 1.02470 0.0068282
2 0.059449      1   0.77125 0.77125 0.0073352
3 0.055141      2   0.71180 0.70290 0.0073237
4 0.020534      3   0.65666 0.65681 0.0072813
5 0.017949      4   0.63613 0.62780 0.0072401
6 0.015796      7   0.57869 0.60454 0.0071988
7 0.011631      8   0.56289 0.56778 0.0071183
8 0.010482      9   0.55126 0.55256 0.0070793
9 0.010000     10   0.54078 0.55428 0.0070839
```

*Figure 6: Initial Wine Quality Decision Tree: Graphical Representation (Left), Cost Complexity Parameter (Right)*

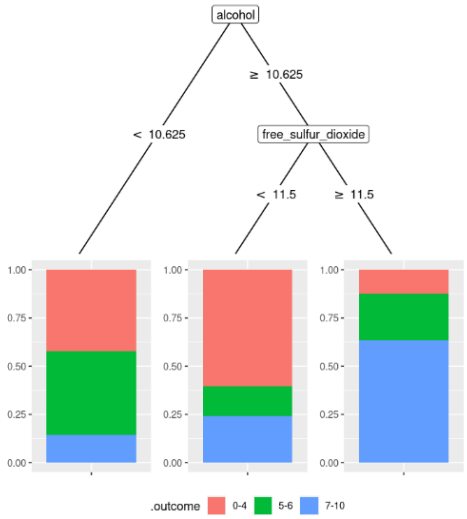
With our aim for this model is for prediction application, we will select to prune or cut the model at the lowest cross-validation error (xerror) point, which is at the 8$^{th}$ position. This is to reduce model complexity and overfitting towards training data. The pruned model is tested with the testing data, producing a confusion matrix (Figure 7), the full output is available in the appendix b.




```
Confusion Matrix and Statistics

          Reference
Prediction 0-4 5-6 7-10
      0-4   37 285   25
      5-6   30 794   91
      7-10   6 413  267

Overall Statistics

               Accuracy : 0.5637
                 95% CI : (0.5413, 0.5858)
    No Information Rate : 0.7659
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2268

 Mcnemar's Test P-Value : <2e-16
```

*Figure 7: Pruned Tree Model: Graphical Representation (Left), Confusion Matrix (Right)*

This model only has an overall testing accuracy of 56.37%, alongside being complex and relatively difficult to interpret. This could be attributed to model overfitting towards the training data. To reduce the model overfitting, we applied k-fold cross-validation (CV) to the tree model. CV is a technique to train and assess a model with different subsets of the training data, generating a more generalized model (Figure 8).




```
Confusion Matrix and Statistics

          Reference
Prediction  0-4  5-6 7-10
      0-4     9   83   37
      5-6    56 1003   97
      7-10    8  406  249

Overall Statistics

               Accuracy : 0.6473
                 95% CI : (0.6256, 0.6686)
    No Information Rate : 0.7659
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2592

 Mcnemar's Test P-Value : <2e-16
```

*Figure 8: 10-fold Cross Validation Tree Model: Graphical Representation (Left), Confusion Matrix (Right)*

This 10-fold CV separates the training data into 10 equally-size group, with one group being used as the testing set while others are for training, reiterating for 10 times. The final model is selected using prediction accuracy as a metric. We performed this technique to pre-prune the model by

optimizing the tree growth. The outputted model (Figure 8, left) is more simplified compared to the post-pruned model. Additionally, the overall test accuracy of this model is higher as the model is more generalized rather than overfitting towards the training data.

While decision tree models have their advantages in simplicity and explainability of the models, their overall accuracy is not optimal towards wine classification applications. This limitation persists despite the accuracy improvements of the CV process. Also, decision tree models are susceptible to overfitting the training data, as demonstrated by the general tree model. To overcome these limitations of decision trees, we will explore a random forest model, which combines multiple decision trees.

### 3.2 Classification Random Forest

Random forest (RF) is an ensemble ML model that combines multiple decision trees into one. It utilizes bootstrap aggregation or bagging to randomly sample a subset of training data to build a decision tree and during each branching process, RF randomly samples a few feature variables to be considered, normally the square root of feature variables (James et al., 2021). This bagging process also mitigates the variance issues of individual decision tree models (ibid). For classification tasks, the final RF model is determined through combined voting of all the generated trees. With the “randomForest” function we created an RF model through the upsampling data with the number of trees of 1,000 trees and the number of parameters of the square root of feature variables. Figure 9 provides the RF training confusion matrix (right) and testing confusion matrix (left).

```
Call:
 randomForest(formula = as.factor(quality_range) ~ .,
data = train_wine_up,      ntree = 1000, mtry = sqrt
(ncol(train_wine_up)))
               Type of random forest: classification
                     Number of trees: 1000
No. of variables tried at each split: 4

        OOB estimate of  error rate: 2.42%
Confusion matrix:
      0-4  5-6 7-10 class.error
0-4  3482    0    0 0.000000000
5-6    14 3250  218 0.066628374
7-10    0   21 3461 0.006031017
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0-4  5-6 7-10
      0-4    12    7    0
      5-6    61 1407  136
      7-10    0   78  247

Overall Statistics

               Accuracy : 0.8552
                 95% CI : (0.8388, 0.8706)
    No Information Rate : 0.7659
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5694

 Mcnemar's Test P-Value : NA
```

*Figure 9: Random Forest Model: Training Output (Left), Testing Confusion Matrix (Right)*

Overall, this model accuracy improves significantly when compared to tree-based models at 85.52%. In addition to the matrix, we plotted out the out of bag error vs Trees, which shows that the error stabilized within our number of ntree choice (Figure 10, left).

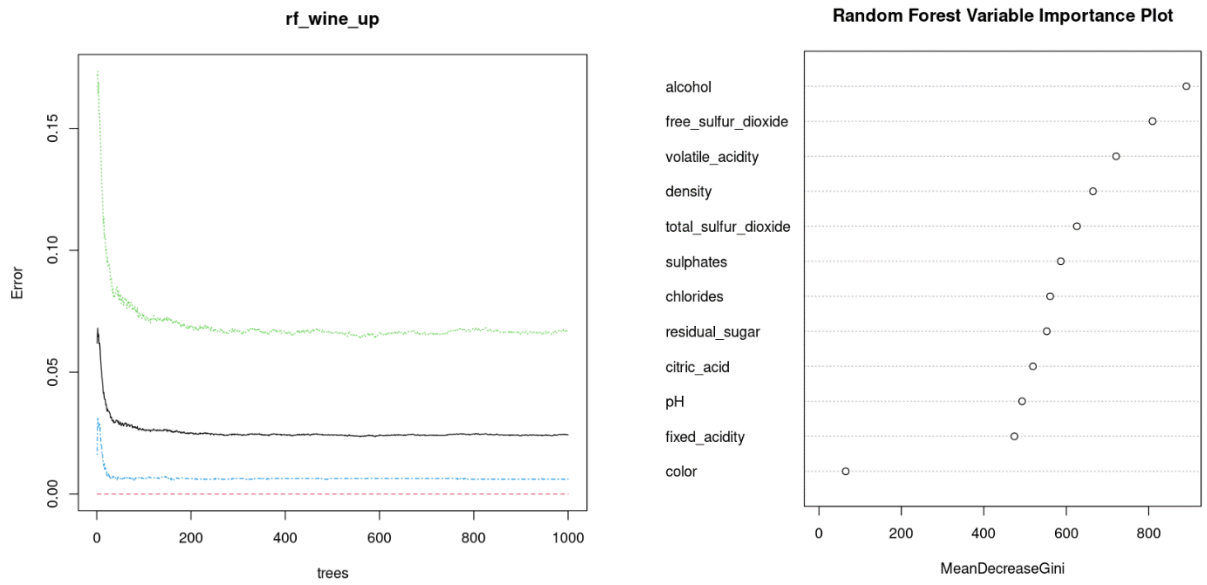



*Figure 10: Random Forest Plots: Out of bag error vs the number of trees (Left), Variable Importance (Right)*

Moreover, we also plotted the variable importance plot, which provides insights into the importance of each predictor variable towards the classification model through mean decrease Gini, Figure 10, right. It is a calculation of the total mean decrease in the Gini index across all trees in the RF model, with higher-importance variables having higher values. From the figure, we found that physiochemical characteristics of alcohol, free sulfur dioxide, and volatile acidity are the three most important variables affecting wine quality. As for the wine color, it is not considered important in the score grading.

## 4. Model Comparison

In this section, we compare three tree-based models using the confusion matrix. We will utilize both the overall model statistics for each model (Table 2) and the statistical parameters of each wine quality class (Table 3). Given the imbalanced classes within the training data, we also consider each class's statistics in addition to overall accuracy. We will evaluate the model based on the assumption that we are focusing on correcting classifying good (7-10 score) quality wine, which is rarer compared to the average (5-6 score) wines, rather than classifying all the wines to their respective categories. Thus, we will give priority to the F1 metrics, which is a harmonic mean of precision and recall, over the balanced accuracy.

Overall, the RF model surpasses single tree models in accuracy, achieving 85.58%, and Kappa score, which measures agreement between predicted and actual classes, compared to single tree models. Additionally, the F1 score for good and average wine surpasses the values of

decision tree models. Nevertheless, the 0-4 score range only achieves a low F1 score. This could be attributed to the heavily imbalanced representation within the original dataset. As for single tree models, the CV model exhibits improvement over the post-pruned model in overall accuracy, while the F1 score is competitive for good and average quality wines. For a post-pruned tree, even though the F1 score is slightly better than the CV tree, the overall model accuracy is slightly better than a coin toss at only 56.37%. From these observations, it can be concluded that the RF model demonstrates superior performance among the models not only in F1 metrics but also in balanced accuracy. This led to it being the preferred model for this application.

*Table 2: Tree-based Model Overall Statistics*

| Model | Method | Accuracy (%) | Kappa* |
|---|---|---|---|
| Single Tree (post-pruned) | rpart | 56.37 | 0.2268 |
| Single Tree (CV) | rpart + caret | 64.73 | 0.2592 |
| Radom Forest | randomForest | 85.52 | 0.5743 |

**Cohen's Kappa coefficient: an agreement probability measurement between interrater, which in this case classifiers, for categorical data. (McHugh M., 2012).*

*Table 3: Tree-based Model Class Statistical Parameters*

| **Statistical Parameters** | **Description** | **Model** | **0-4 Score (Bad Wine)** | **5-6 Score (Average Wine)** | **7-10 Score (Good Wine)** |
|---|---|---|---|---|---|
| Sensitivity | Measurement of model ability to predict true positive classes. | Single Tree (post-pruned) | 0.50685 | 0.5322 | 0.6971 |
| | | Single Tree (CV) | 0.12329 | 0.6723 | 0.6501 |
| | | Radom Forest | 0.164384 | 0.9430 | 0.6449 |
| Specificity | Measurement of model ability to predict true negative classes | Single Tree (post-pruned) | 0.83467 | 0.7346 | 0.7323 |
| | | Single Tree (CV) | 0.93600 | 0.6645 | 0.7355 |
| | | Radom Forest | 0.996267 | 0.5680 | 0.9502 |
| Balanced Accuracy | An average of sensitivity and specificity. | Single Tree (post-pruned) | 0.67076 | 0.6334 | 0.7147 |
| | | Single Tree (CV) | 0.52964 | 0.6684 | 0.6928 |
| | | Radom Forest | 0.580325 | 0.7555 | 0.7975 |
| Precision | Proportion of true positives over true positives and false positives | Single Tree (post-pruned) | 0.10663 | 0.8678 | 0.3892 |
| | | Single Tree (CV) | 0.06977 | 0.8676 | 0.3756 |
| | | Radom Forest | 0.631579 | 0.8772 | 0.7600 |
| Recall | Proportion of true positives over classified positives | Single Tree (post-pruned) | 0.50685 | 0.5322 | 0.6971 |
| | | Single Tree (CV) | 0.12329 | 0.6723 | 0.6501 |
| | | Radom Forest | 0.164384 | 0.9430 | 0.6449 |

| | | | | | |
|---|---|---|---|---|---|
| F1 | Harmonic mean of recall and precision | Single Tree (post-pruned) | 0.17619 | 0.6597 | 0.4995 |
| | | Single Tree (CV) | 0.08911 | 0.7576 | 0.4761 |
| | | Radom Forest | 0.260870 | 0.9289 | 0.6977 |

# 5. Result and Conclusion

## 5.1 Result and Model Limitations

The analysis indicates that the RF model is the optimal choice for wine quality classification among the tree-based models with an accuracy score of 85.52%. Moreover, the RF model demonstrates strong performance in its class statistical scores, especially F1-score and balanced accuracy, despite the imbalanced dataset, where average wine dominates. From, the RF variable importance plot, it can be identified that the three most important variables are alcohol, free sulfur dioxide, and volatile acidity, accordingly, while color does not have a significant impact on the model. This outcome matches with our initial hypothesis formulated during EDA, apart from chlorides and density, which does not make into the top three high-importance variables.

Despite the high accuracy, there are limitations when using this model. Firstly, the collected ratings are imbalanced, with the underrepresentation of good and bad wine qualities, hindering the model generalization from these minority classes. Furthermore, despite grouping the qualities into three ranges to accommodate for all possible scores, the model is only applicable to predicting wine that has a score between 3 and 9, as that is our training data. Additionally, our use of upsampling to address this issue and enhance the sensitivity and specificity towards each class may result in model overfitting towards the minority classes. Thus, the model performance will decrease when applied to unseen data.

## 5.2 Conclusion and Future Improvements

In conclusion, this report explores the classification of red and white wine data sets into good, average, and bad quality ranges with tree-based models. Through exploratory data analytics, we identified that these three categories of wine are imbalanced, as the majority of the observations are average-rated wines at 76.56%. This leads to our choice of using the upsampling method to increase the training data for good and bad quality wine. We generated a decision tree model and a random forest model for this wine classification application. From the comparison, we select a random forest model for this application as it achieves the highest overall accuracy and F1 score for each of the classes.

To improve this model, we suggest the collection of additional data on red and white wine physiochemical characteristics and their respective quality score. This approach aims to capture

each wine type's characteristics to improve their representation within the data. Alternatively, with the existing dataset, we proposed the use of the Synthetic Minority Over-sampling Technique (SMOTE) to generate minority class data over the use of the available caret resampling methods. This is to combat biased model resulted from minority class oversampling.

## Bibliography

Bhardwaj, P., Tiwari, P., Olejar Jr, K., Parr, W., & Kulasiri, D. (2022). A machine learning application in wine quality prediction. *Machine Learning with Applications*, *8*, 100261.

Candiago, S., Tscholl, S., Bassani, L., Fraga, H., & Egarter Vigl, L. (2022). A geospatial inventory of regulatory information for wine protected designations of origin in Europe. *Scientific Data*, *9*(1), 394.

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., & Reis, J. (2009a). Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, *47*(4), 547–553. https://doi.org/10.1016/j.dss.2009.05.016

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., & Reis, J. (2009b). Wine quality. UCI Machine Learning Repository. Available at: https://archive.ics.uci.edu/dataset/186/wine+quality (Accessed: 05 February 2024).

Diez, D.M., Barr, C.D. and Çetinkaya-Rundel, M. (2019) *OpenIntro Statistics*. 4th edn. Boston, MA, USA.

FAOSTAT (2023) *Crops and livestock products (Production)*, *FAOSTAT*. Available at: https://www.fao.org/faostat/en/#data (Accessed: 05 February 2024).

International Organisation of Vine and Wine (OIV) (2022) *Compendium of international methods of wine and must analysis*, *OIV*. Available at: https://www.oiv.int/standards/compendium-of-international-methods-of-wine-and-must-analysis (Accessed: 06 February 2024).

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An introduction to statistical learning: with Applications in R.* (2nd ed.). New York: springer.

McHugh M. L. (2012). Interrater reliability: the kappa statistic. *Biochemia medica*, *22*(3), 276–282.

Pangborn, R. M., Berg, H. W., & Hansen, B. (1963). The influence of color on discrimination of sweetness in dry table-wine. *The american journal of psychology*, *76*(3), 492-495.

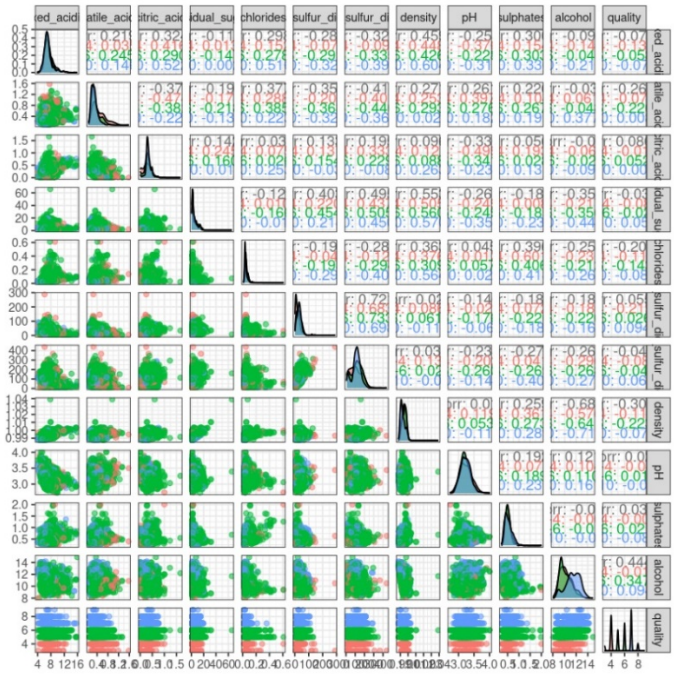
Parr, W. V., Geoffrey White, K., & Heatherbell, D. A. (2003). The nose knows: Influence of colour on perception of wine aroma. *Journal of wine research*, *14*(2-3), 79-101.

Spence, C., Levitan, C. A., Shankar, M. U., & Zampini, M. (2010). Does food color influence taste and flavor perception in humans?. *Chemosensory perception*, *3*, 68-84.

Soleas, G. J., Diamandis, E. P., & Goldberg, D. M. (1997). Wine as a biological fluid: history, production, and role in disease prevention. *Journal of clinical laboratory analysis*, *11*(5), 287–313. https://doi.org/10.1002/(SICI)1098-2825(1997)11:5<287::AID-JCLA6>3.0.CO;2-4

# Appendix

a. Wine Data Matix of Scatter Plots




b. Post- Pruned Decision Tree (left) vs CV Tree (right) Confusion Matrix and Statistics

```
Confusion Matrix and Statistics

          Reference
Prediction 0-4 5-6 7-10
      0-4   37 285   25
      5-6   30 794   91
      7-10   6 413  267

Overall Statistics

               Accuracy : 0.5637
                 95% CI : (0.5413, 0.5858)
    No Information Rate : 0.7659
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2268

 Mcnemar's Test P-Value : <2e-16

Statistics by Class:

                     Class: 0-4 Class: 5-6 Class: 7-10
Sensitivity             0.50685     0.5322      0.6971
Specificity             0.83467     0.7346      0.7323
Pos Pred Value          0.10663     0.8678      0.3892
Neg Pred Value          0.97751     0.3243      0.9081
Precision               0.10663     0.8678      0.3892
Recall                  0.50685     0.5322      0.6971
F1                      0.17619     0.6597      0.4995
Prevalence              0.03747     0.7659      0.1966
Detection Rate          0.01899     0.4076      0.1371
Detection Prevalence    0.17813     0.4697      0.3522
Balanced Accuracy       0.67076     0.6334      0.7147
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0-4  5-6 7-10
      0-4     9   83   37
      5-6    56 1003   97
      7-10    8  406  249

Overall Statistics

               Accuracy : 0.6473
                 95% CI : (0.6256, 0.6686)
    No Information Rate : 0.7659
    P-Value [Acc > NIR] : 1

                  Kappa : 0.2592

 Mcnemar's Test P-Value : <2e-16

Statistics by Class:

                     Class: 0-4 Class: 5-6 Class: 7-10
Sensitivity             0.12329     0.6723      0.6501
Specificity             0.93600     0.6645      0.7355
Pos Pred Value          0.06977     0.8676      0.3756
Neg Pred Value          0.96482     0.3826      0.8957
Precision               0.06977     0.8676      0.3756
Recall                  0.12329     0.6723      0.6501
F1                      0.08911     0.7576      0.4761
Prevalence              0.03747     0.7659      0.1966
Detection Rate          0.00462     0.5149      0.1278
Detection Prevalence    0.06622     0.5934      0.3403
Balanced Accuracy       0.52964     0.6684      0.6928
```

c. Random Forest Confusion Matrix and Statistics

```
Confusion Matrix and Statistics

          Reference
Prediction  0-4  5-6 7-10
      0-4    12    7    0
      5-6    61 1407  136
      7-10    0   78  247

Overall Statistics

               Accuracy : 0.8552
                 95% CI : (0.8388, 0.8706)
    No Information Rate : 0.7659
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5694

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: 0-4 Class: 5-6 Class: 7-10
Sensitivity            0.164384     0.9430      0.6449
Specificity            0.996267     0.5680      0.9502
Pos Pred Value         0.631579     0.8772      0.7600
Neg Pred Value         0.968377     0.7529      0.9162
Precision              0.631579     0.8772      0.7600
Recall                 0.164384     0.9430      0.6449
F1                     0.260870     0.9089      0.6977
Prevalence             0.037474     0.7659      0.1966
Detection Rate         0.006160     0.7223      0.1268
Detection Prevalence   0.009754     0.8234      0.1668
Balanced Accuracy      0.580325     0.7555      0.7975
```