Statistical Analysis of Wine Quality

Table of Contents

Table of Figures

1.1 Introduction

The wine industry is highly competitive, and implementing statistical analysis to generate data-driven insight can provide an edge over competitors. This report aims to analyze the various physicochemical properties and quality of red and white variants of the Portuguese Vinho Verde wine and then assess if predictions can be made to improve overall quality and maximize success.

1.1.1 Objectives

- Compare the physicochemical properties of both red and white wine to uncover how they affect the quality.
- Test hypotheses that can provide actionable insights for improving wine quality.
- Make predictions of wine quality based on the physicochemical properties and wine type.

1.1.2 Data Overview

Cortez et al. (2009) created this dataset which consists of two subsets; red wines and white wines.

```
# Importing the data set
red_wine <- read_excel("winequality-red.xlsx")
white_wine <- read_excel("winequality-white.xlsx")
```

Figure 2.1 Loading the data from Excel in R.

Both data consist of 11 physicochemical properties as the input variables and quality as the output.

Table 2.1 Description of the physicochemical properties of wine

PHYSIOCHEMICAL PROPERTIES	DESCRIPTION
FIXED ACIDITY	Non-volatile acids in wine
VOLATILE ACIDITY	Acetic acid concentration
CITRIC ACID	Adds freshness to wine
RESIDUAL SUGAR	Remaining sugar after fermentation
CHLORIDES	Salt concentration
FREE SULFUR DIOXIDE	Unbound sulfur dioxide preventing microbial growth
TOTAL SULFUR DIOXIDE	Combined free and bound sulfur dioxide
DENSITY	Wine's mass-to-volume ratio
PH	Acidity levels
SULPHATE	Added for flavor and stability
ALCOHOL	Ethanol content

1.2 Exploratory Data Analysis (EDA)

Exploring and analysing the features in a dataset is essential for gaining a proper understanding and developing a foundation for statistical analysis. This process will include data cleaning and visualizations that provide valuable insights.

1.2.1 Data Understanding and Cleaning

Missing Values: No missing values were present in the dataset.

```
# checking for missing values
sum(is.na(red_wine))
```

```
## [1] 0
```

```
sum(is.na(white_wine))
```

```
## [1] 0
```

Figure 2.2 R code and output checking for missing values in red and white wine data

Duplicates: There were 240 and 937 duplicated rows in the red and white wine data respectively. The duplicated rows were removed due to errors due to bias and redundancies.

```
# Checking for duplicates
sum(duplicated(red_wine))
```

```
## [1] 240
```

```
sum(duplicated(white_wine))
```

```
## [1] 937
```

```
# Removing duplicates
red_wine <- red_wine[!duplicated(red_wine), ]
white_wine <- white_wine[!duplicated(white_wine), ]
```

Figure 2.3 R code and output for handling duplicated values in red and white wine data

Merging: Since both red and white wine have the same variables, the datasets were merged by introducing a wine-type column to aid easy analysis.

```
# Merging the data frames for easier analysis
red_wine <- mutate(red_wine, wine_type = as.factor("Red"))
white_wine <- mutate(white_wine, wine_type = as.factor("White"))
wine_data <- bind_rows(red_wine, white_wine)
```

Figure 2.4 R code merging the red and white data

Converting Data Type: Quality scores were converted to ordinal categories from 3-9.

```
#converting quality to ordinal categorical
wine_data$quality <- factor(wine_data$quality, ordered = TRUE)
```

Figure 2.5 R code converting the Quality column to ordered categories

Structure: The new data frame consists of 13 variables, 11 of which are continuous while the remaining two are categorical variables.

```r
# checking the structure of the wine_data data frame
str(wine_data)
```

```
## tibble [5,320 x 13] (S3: tbl_df/tbl/data.frame)
##  $ fixed acidity       : num [1:5320] 7.4 7.8 7.8 11.2 7.4 7.9 7.3 7.8 7.5 6.7 ...
##  $ volatile acidity    : num [1:5320] 0.7 0.88 0.76 0.28 0.66 0.6 0.65 0.58 0.5 0.58 ...
##  $ citric acid         : num [1:5320] 0 0 0.04 0.56 0 0.06 0 0.02 0.36 0.08 ...
##  $ residual sugar      : num [1:5320] 1.9 2.6 2.3 1.9 1.8 1.6 1.2 2 6.1 1.8 ...
##  $ chlorides           : num [1:5320] 0.076 0.098 0.092 0.075 0.075 0.069 0.065 0.073 0.071 0.097 .
##  $ free sulfur dioxide : num [1:5320] 11 25 15 17 13 15 15 9 17 15 ...
##  $ total sulfur dioxide: num [1:5320] 34 67 54 60 40 59 21 18 102 65 ...
##  $ density             : num [1:5320] 0.998 0.997 0.997 0.998 0.998 ...
##  $ pH                  : num [1:5320] 3.51 3.2 3.26 3.16 3.51 3.3 3.39 3.36 3.35 3.28 ...
##  $ sulphates           : num [1:5320] 0.56 0.68 0.65 0.58 0.56 0.46 0.47 0.57 0.8 0.54 ...
##  $ alcohol             : num [1:5320] 9.4 9.8 9.8 9.8 9.4 9.4 10 9.5 10.5 9.2 ...
##  $ quality             : Ord.factor w/ 7 levels "3"<"4"<"5"<"6"<..: 3 3 3 4 3 3 5 5 3 3 ...
##  $ wine_type           : Factor w/ 2 levels "Red","White": 1 1 1 1 1 1 1 1 1 1 ...
```

The red wine had 1359 rows and 12 columns The white wine had 3961 rows and 12 columns The new data frame (wine_data) has 5320 row and 13 columns. The quality column is now a categorical variable with 7 levels from 3-9 The last column represents the wine type, red or white.

Figure 2.6 R code and output showing the structure of the merged data

1.2.2 Summary Statistics

From initial summary statistics, the wine data has a total of 5,320 observations, with 1,359 red wines and 3,961 white wines. The quality of the wines is skewed towards the mid-range values of 5 and 6, with very few wines with excellent or poor ratings.

```r
# Summary statistics
summary(wine_data)
```

```
##  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.:0.2400   1st Qu.: 1.800
##  Median : 7.000   Median :0.3000   Median :0.3100   Median : 2.700
##  Mean   : 7.215   Mean   :0.3441   Mean   :0.3185   Mean   : 5.048
##  3rd Qu.: 7.700   3rd Qu.:0.4100   3rd Qu.:0.4000   3rd Qu.: 7.500
##  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.: 16.00      1st Qu.: 74.0        1st Qu.:0.9922
##  Median :0.04700   Median : 28.00      Median :116.0        Median :0.9947
##  Mean   :0.05669   Mean   : 30.04      Mean   :114.1        Mean   :0.9945
##  3rd Qu.:0.06600   3rd Qu.: 41.00      3rd Qu.:153.2        3rd Qu.:0.9968
##  Max.   :0.61100   Max.   :289.00      Max.   :440.0        Max.   :1.0390
##
##        pH          sulphates         alcohol      quality  wine_type
##  Min.   :2.720   Min.   :0.2200   Min.   : 8.00   3:  30   Red  :1359
##  1st Qu.:3.110   1st Qu.:0.4300   1st Qu.: 9.50   4: 206   White:3961
##  Median :3.210   Median :0.5100   Median :10.40   5:1752
##  Mean   :3.225   Mean   :0.5334   Mean   :10.55   6:2323
##  3rd Qu.:3.330   3rd Qu.:0.6000   3rd Qu.:11.40   7: 856
##  Max.   :4.010   Max.   :2.0000   Max.   :14.90   8: 148
##                                                   9:   5
```

From the summary statistics: There are 30 observations with a rating of 3 There are 206 observations with a rating of 4 There are 1752 observations with a rating of 5 There are 2323 observations with a rating of 6 There are 856 observations with a rating of 7 There are 148 observations with a rating of 8 There are 5 observations with a rating of 9

Figure 2.7 Summary statistics of wine data

```r
# summary statistics using the skim function
wine_data %>%
  group_by(wine_type) %>%
  skim()
```

Table 1: Data summary

Name	Piped data
Number of rows	5320
Number of columns	13
_____	
Column type frequency:	
factor	1
numeric	11
_____	
Group variables	wine_type

Variable type: factor

skim_variable	wine_type	n_missing	complete_rate	ordered	n_unique	top_counts
quality	Red	0	1	TRUE	6	5: 577, 6: 535, 7: 167, 4: 53
quality	White	0	1	TRUE	7	6: 1788, 5: 1175, 7: 689, 4: 153

Figure 2.8 Summary statistics using skimr library

1.2.3 Distributions

Quality: Figure 2.9 shows that the majority of red and white wines have a quality score of 5 & 6.

```r
# Plotting the percentage distribution
ggplot(quality_percentage, aes(x = quality, y = percentage, fill = wine_type)) +
  geom_col(position = "dodge") +
  labs(title = "Percentage Distribution of Wine Quality by Type") +
  scale_fill_manual(values = c("Red" = "red", "White" = "lightblue")) +  theme_calc()
```

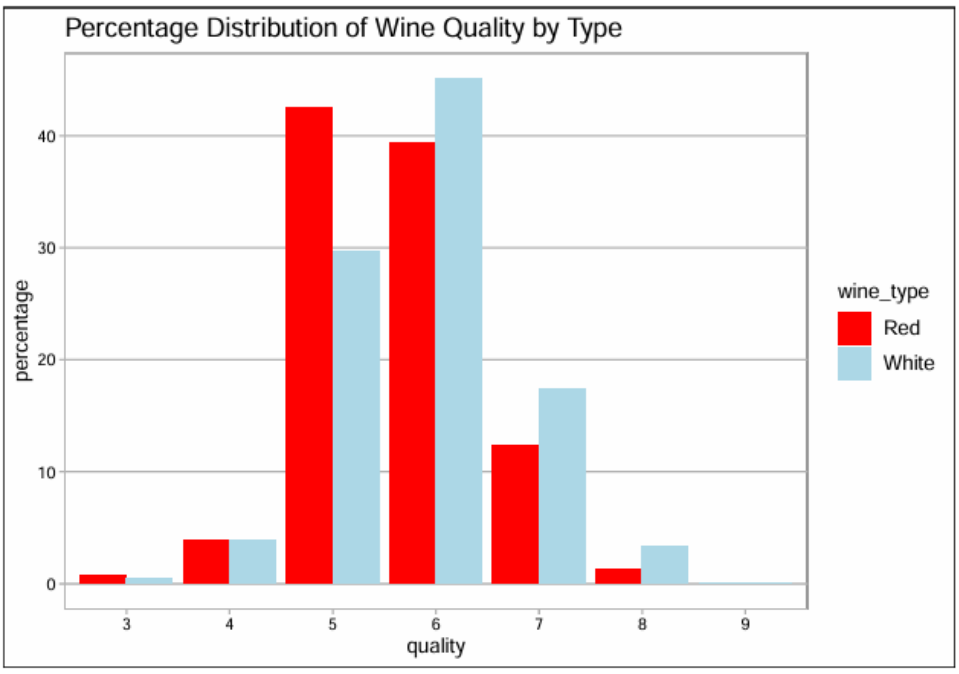


Figure 2.9 Clustered column chart showing the percentage distribution of wine quality by type

For ease of analysis, four functions were created to carry out a basic distribution analysis of all the physicochemical properties.

Plot_boxplot: for plotting boxplots of the property by the two wine types.

```
# creating a function to plot box plot by wine type
plot_boxplot <- function(data, column_name) {
  ggplot(data, aes(x = wine_type, y = .data[[column_name]], fill = wine_type)) +
    geom_boxplot() +
    labs(title = paste("Boxplot of", column_name, "by Wine Type")) +
    scale_fill_manual(values = c("Red" = "red", "White" = "lightblue")) +
    theme_calc() +
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
}
```

Figure 2.10 R code creating a function to plot a boxplot of physicochemical properties by wine type

Plot_boxplot_quality: for plotting boxplots of the property by the two wine types and quality.

```
# creating a function to plot box plot by quality and wine type
plot_boxplot_quality <- function(data, column_name) {
  ggplot(data, aes(x = quality, y = .data[[column_name]], fill = wine_type)) +
    geom_boxplot() +
    labs(title = paste("Boxplot of", column_name, "by quality and Wine Type")) +
    scale_fill_manual(values = c("Red" = "red", "White" = "lightblue")) +
    theme_calc() +
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
}
```

Figure 2.11 R code creating a function to plot a boxplot of physicochemical properties by wine type and quality

Plot_density_plot: for plotting density plots based on wine type.

```
# creating a function to plot density plot
plot_density_plot <- function(data, column_name) {
  ggplot(data, aes(x = .data[[column_name]], fill = wine_type)) +
    geom_density(alpha = 0.7) +
    labs(title = paste("Density plot of", column_name, "by Wine Type")) +
    scale_fill_manual(values = c("Red" = "red", "White" = "lightblue")) +
    theme_calc() +
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
}
```

Figure 2.12 R code creating a function to plot a density plot of physicochemical properties by wine type

Normality_tester: Plots a QQ plot and performs the Shapiro-Wilks test of normality

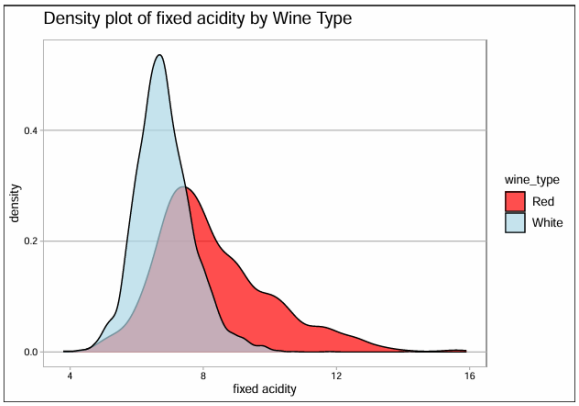
```r
# Function to perform normality test.
# Creating a function to plot QQ plot and perform Shapiro-Wilks test
normality_tester <- function(data, column_name, color, wine_type) {
  column_data <- data[[column_name]]

  # QQ plot
  qqplot <- ggplot(data, aes(sample = column_data)) +
    stat_qq(color = color) +
    stat_qq_line(color = "black") +
    labs(
      title = paste("QQ Plot for", column_name, "(", wine_type, ")"),
      x = "Theoretical",
      y = "Sample"
    ) +
    theme_calc()+
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
  print(qqplot)

  #Shapiro-Wilks test
  shapiro_result <- shapiro.test(column_data)
  return(shapiro_result)
}
```

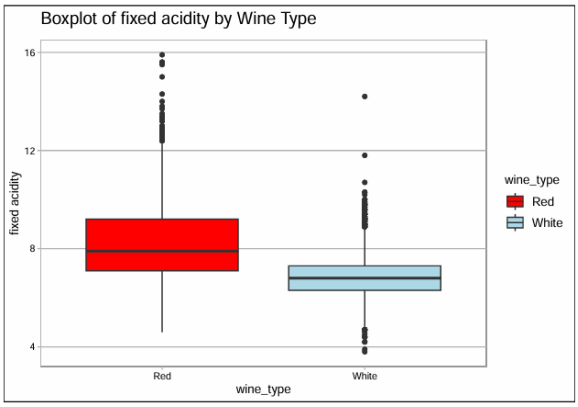
Figure 2.13 R code to create a function that plots a QQ plot and performs Shapiro-Wilk's test

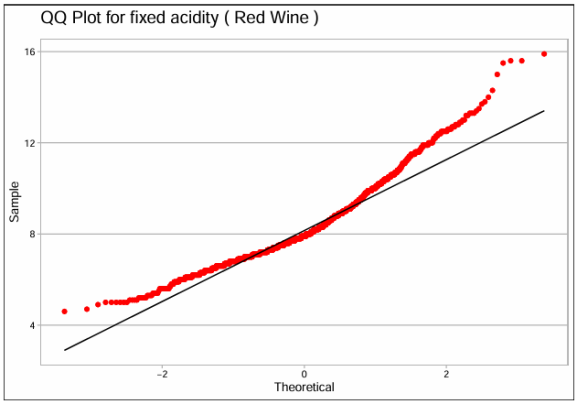
Fixed Acidity: Red wine appears to have a higher median for fixed acidity and they both have outliers. They both appear to have a non-normal distribution with a p-value less than 0.05.

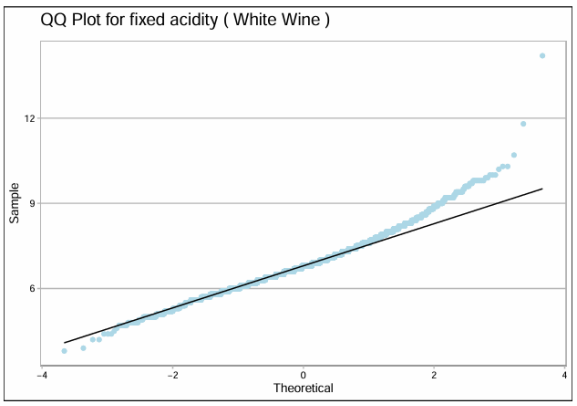
```r
#fixed acidity
plot_density_plot(wine_data, 'fixed acidity')
```

```r
plot_boxplot(wine_data, 'fixed acidity')
```







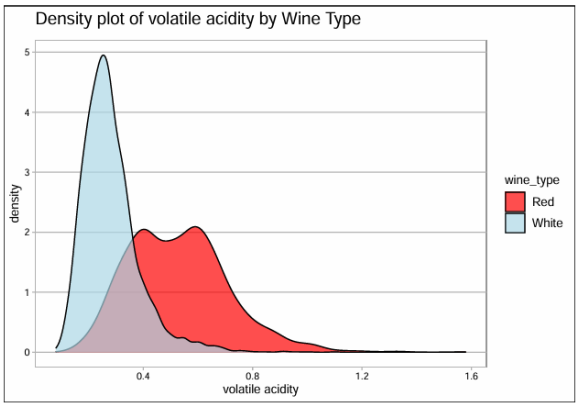


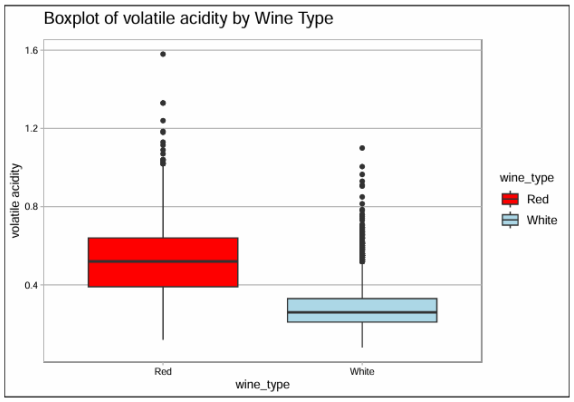
```
Shapiro-Wilk normality test

data:  column_data
W = 0.94684, p-value < 2.2e-16
```

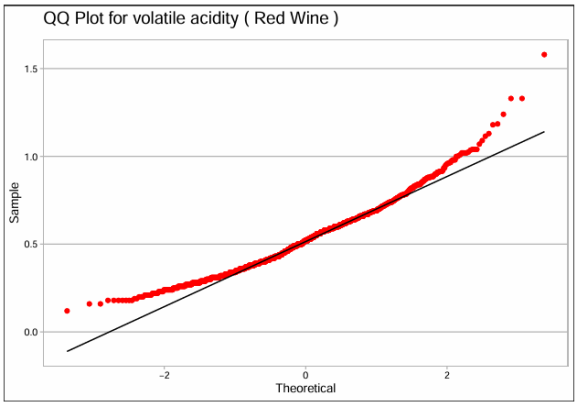
```
Shapiro-Wilk normality test

data:  column_data
W = 0.97418, p-value < 2.2e-16
```

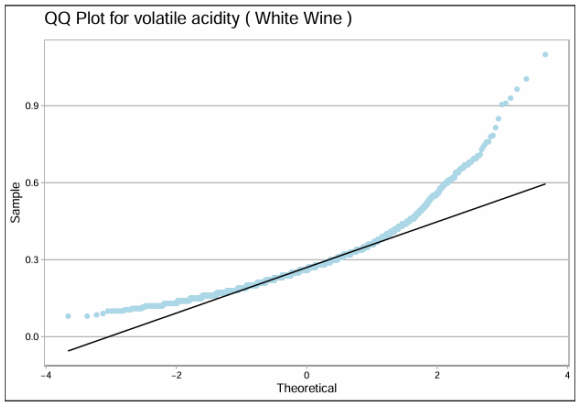
Figure 2.14 Distribution and Normality test of fixed acidity for red and white wine

Volatile Acidity: Median volatile acidity is significantly higher in red wines, the normality test shows that Volatile acidity has a non-normal distribution across wine types







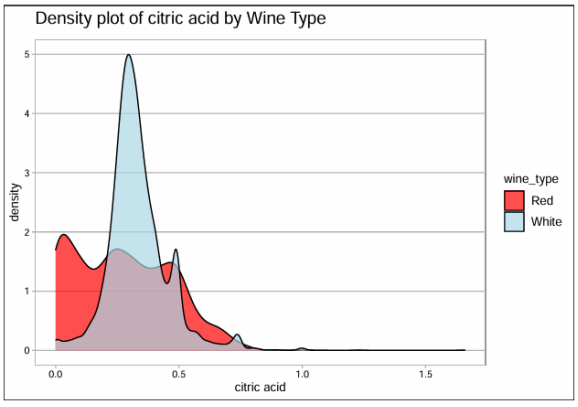


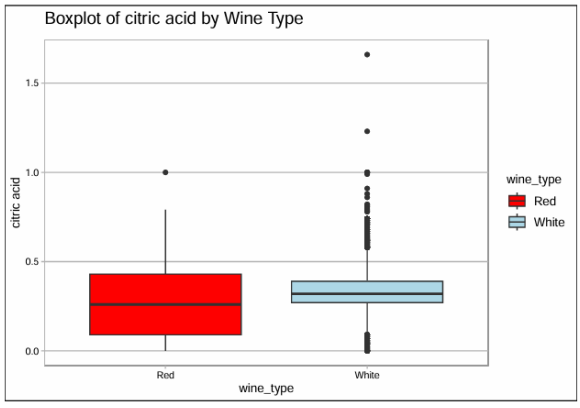

```
Shapiro-Wilk normality test

data:  column_data
W = 0.97018, p-value = 3.931e-16
```

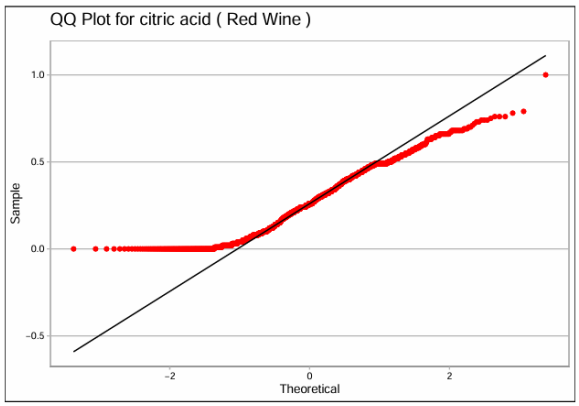
```
Shapiro-Wilk normality test

data:  column_data
W = 0.89753, p-value < 2.2e-16
```

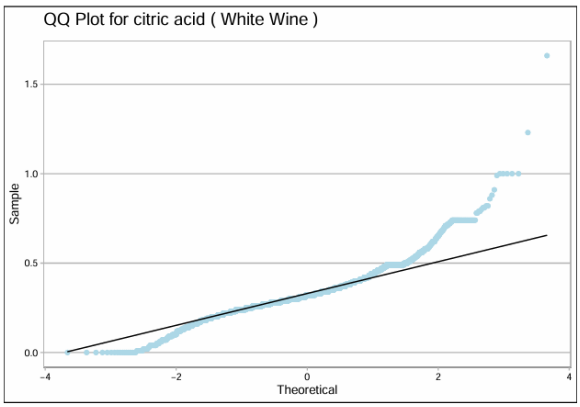
Figure 2.15 Figure 2.14 Distribution and Normality test of volatile acidity for red and white wine

Citric Acid: White wine shows higher median citric acid levels than red wine. White wine appears right skewed while red wine appears uniformly distributed. They are both non-normally distributed.







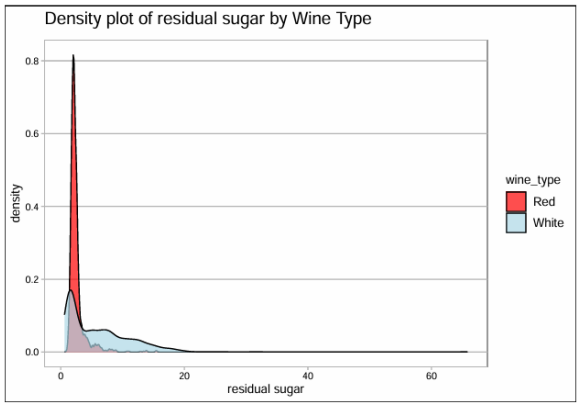


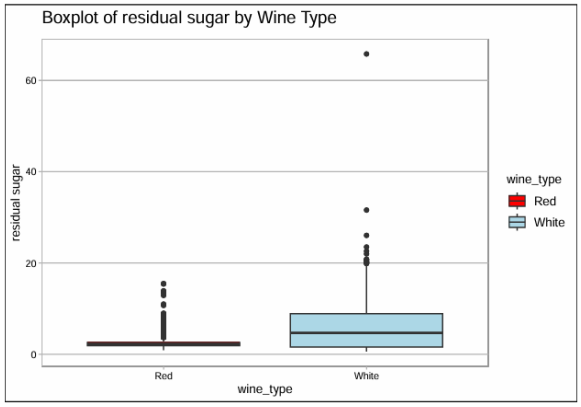

```
Shapiro-Wilk normality test

data:  column_data
W = 0.95552, p-value < 2.2e-16
```

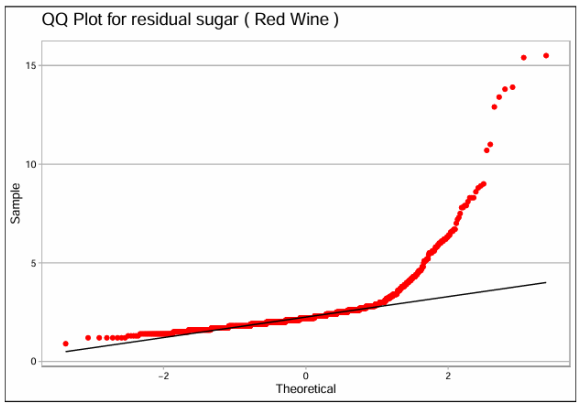
```
Shapiro-Wilk normality test

data:  column_data
W = 0.92054, p-value < 2.2e-16
```

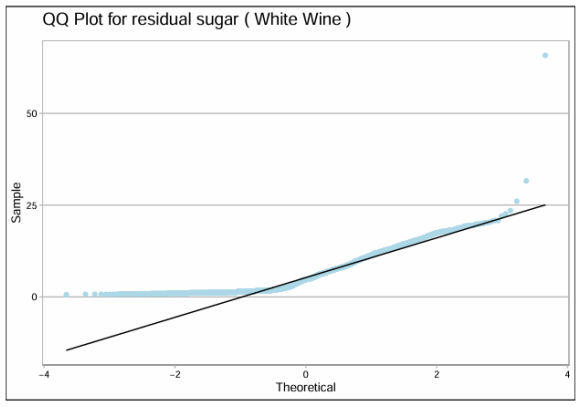
Figure 2.16 Figure 2.14 Distribution and Normality test of citric acid for red and white wine

Residual Sugar: White wines show significantly higher residual sugar levels with a wider spread. They both appear right skewed and non-normal with p-values less than 0.05.







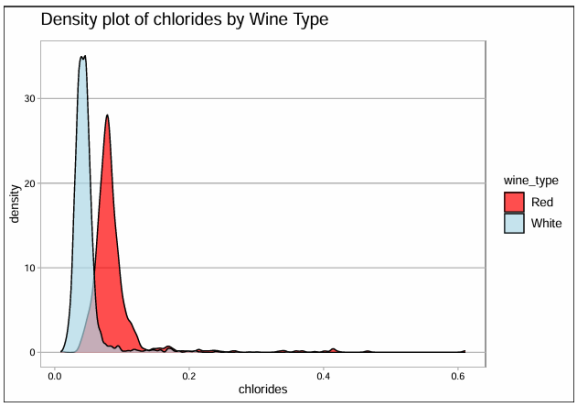


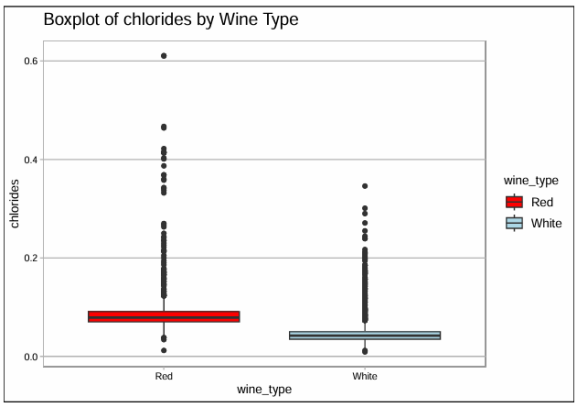

```
Shapiro-Wilk normality test

data:  column_data
W = 0.57673, p-value < 2.2e-16
```

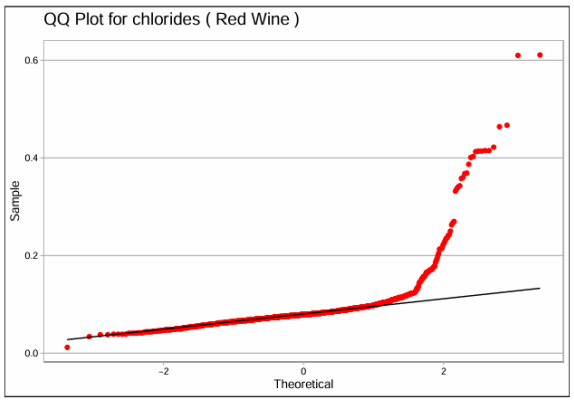
```
Shapiro-Wilk normality test

data:  column_data
W = 0.86654, p-value < 2.2e-16
```

Figure 2.17 Figure 2.14 Distribution and Normality test of residual sugar for red and white wine

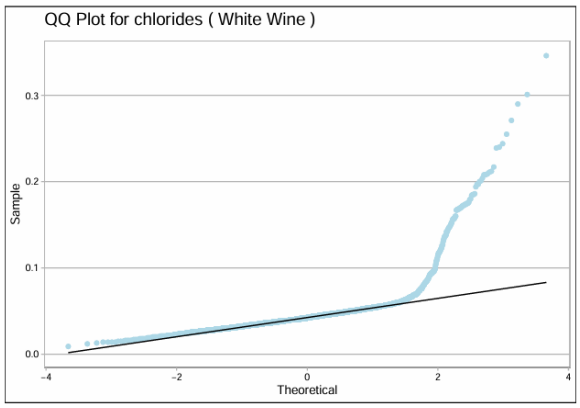
Chlorides: Outliers are pretty apparent in chlorides for both wine types, red wines show a slightly higher median value. The p-values are less than 0.05, therefore they have a non-normal distribution.







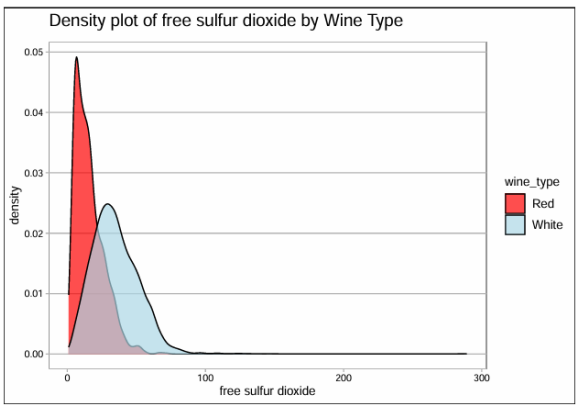

```
Shapiro-Wilk normality test

data:  column_data
W = 0.48448, p-value < 2.2e-16
```

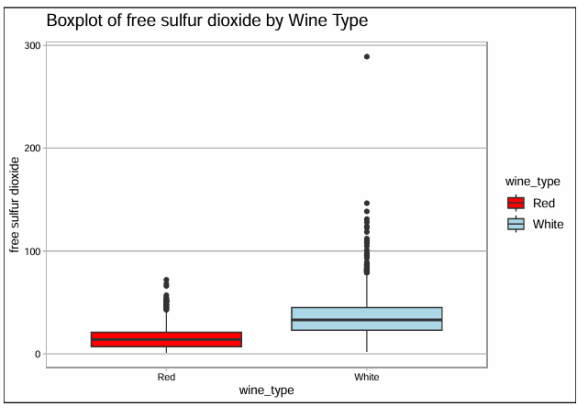


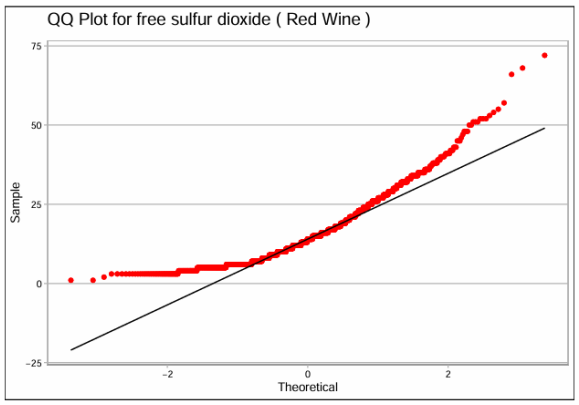

```
Shapiro-Wilk normality test

data:  column_data
W = 0.57789, p-value < 2.2e-16
```

Figure 2.18 Figure 2.14 Distribution and Normality test of chlorides for red and white wine

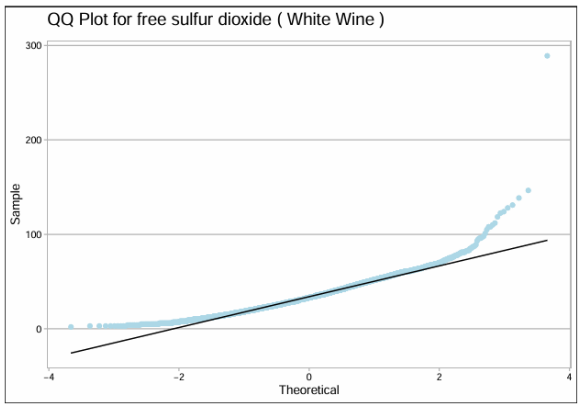
Free Sulfur Dioxide: White wines have significantly higher median levels and outliers. P < 0.05 for both wine types.







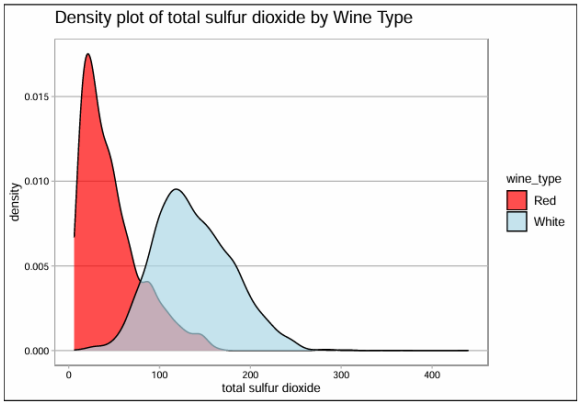

```
Shapiro-Wilk normality test

data:  column_data
W = 0.90323, p-value < 2.2e-16
```

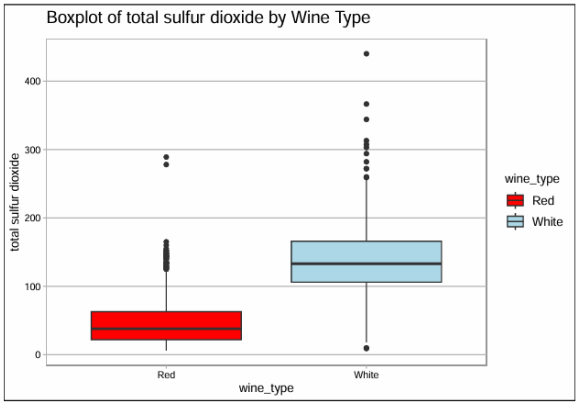


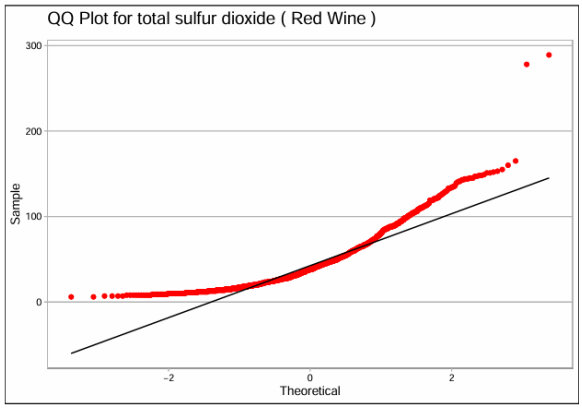

```
Shapiro-Wilk normality test

data:  column_data
W = 0.93339, p-value < 2.2e-16
```

Figure 2.19 Figure 2.14 Distribution and Normality test of free sulfur dioxide for red and white wine

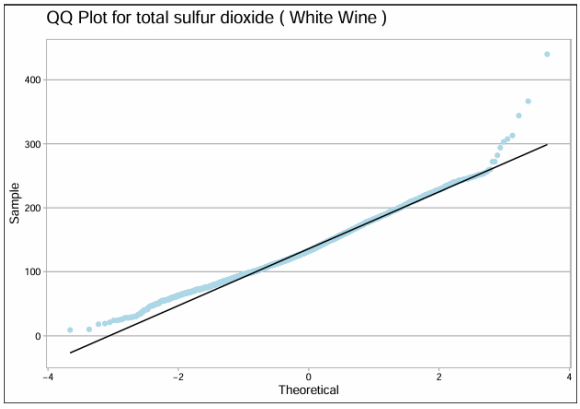
Total Sulfur Dioxide: White wine has a significantly higher median. Both red and white wine have p-values less than 0.05, therefore they are not normally distributed.







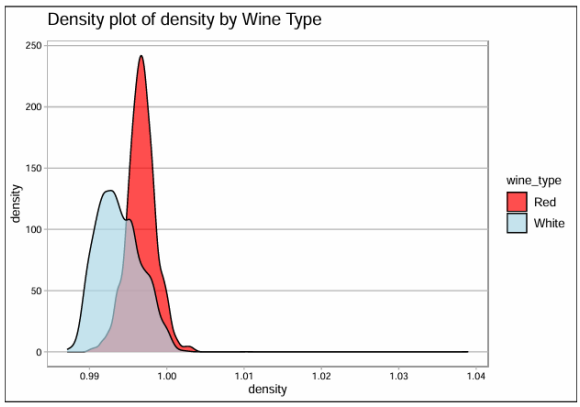

```
Shapiro-Wilk normality test

data:  column_data
W = 0.87169, p-value < 2.2e-16
```

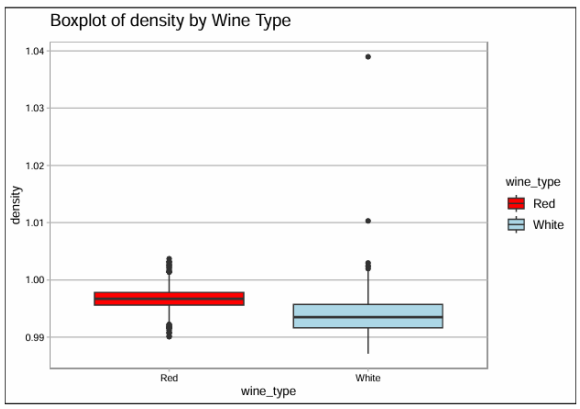


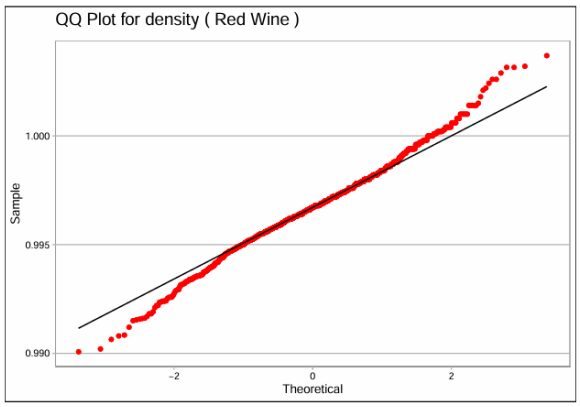

```
Shapiro-Wilk normality test

data:  column_data
W = 0.98638, p-value < 2.2e-16
```

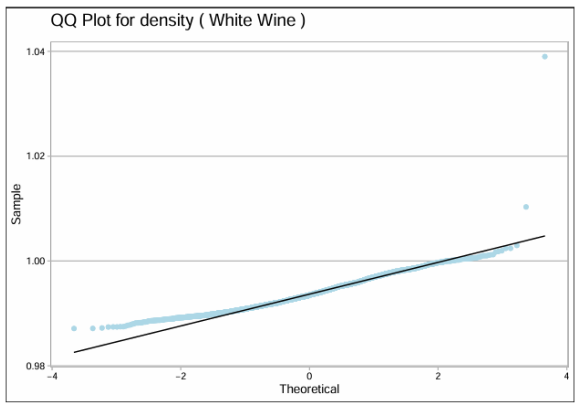
Figure 2.20 Figure 2.14 Distribution and Normality test of total sulfur dioxide for red and white wine

Density: White wines show higher median density QQ plots show almost normality but fail Shapiro's test with P < 0.05.







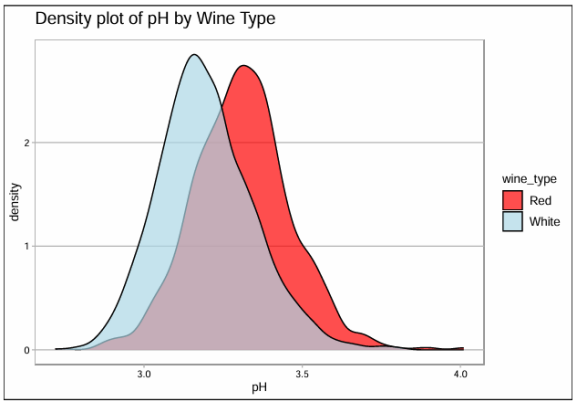


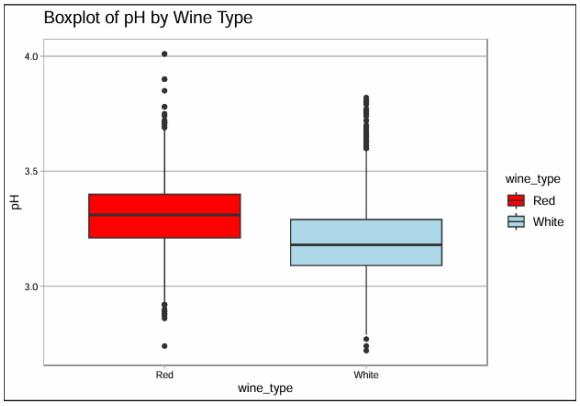

```
Shapiro-Wilk normality test

data:  column_data
W = 0.99239, p-value = 1.804e-06
```

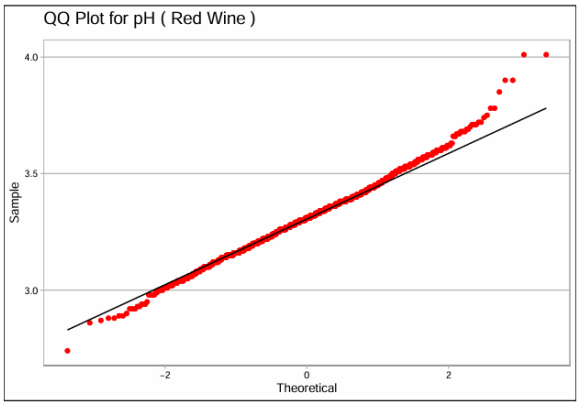
```
Shapiro-Wilk normality test

data:  column_data
W = 0.94739, p-value < 2.2e-16
```

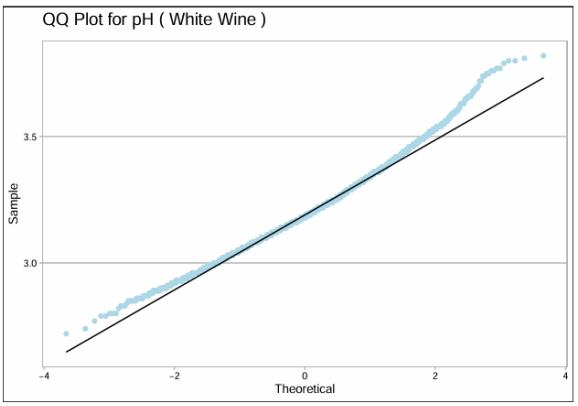
Figure 2.21 Figure 2.14 Distribution and Normality test of density for red and white wine

pH: Red wines have higher median, density plots for both wines appear like a bell curve. The normality test indicates that P < 0.05, therefore they have a non-normal distribution. A log transformation will be ideal to make it approximately normal.







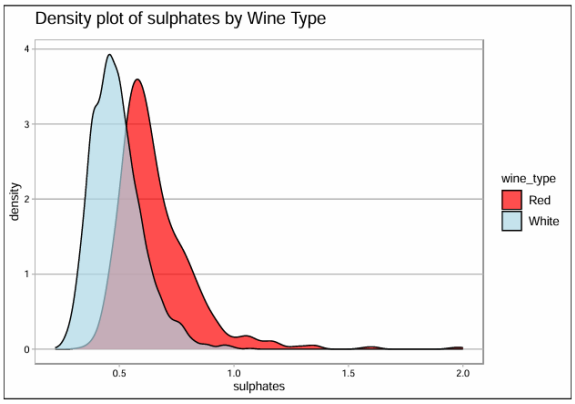


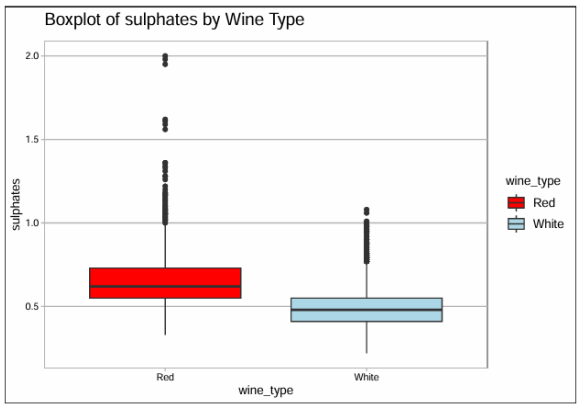

```
Shapiro-Wilk normality test

data:  column_data
W = 0.99272, p-value = 3.082e-06
```

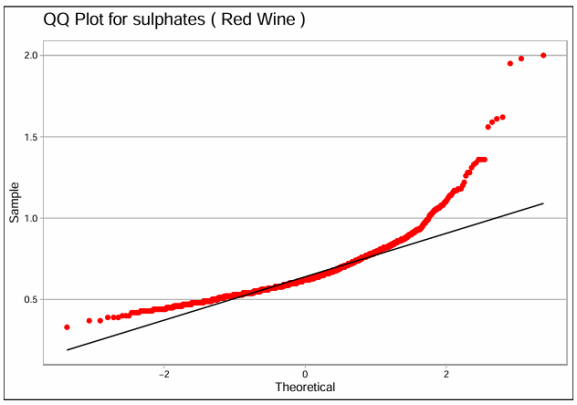
```
Shapiro-Wilk normality test

data:  column_data
W = 0.98817, p-value < 2.2e-16
```

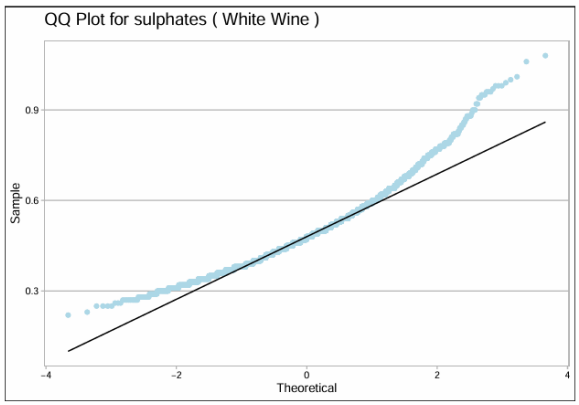
Figure 2.22 Figure 2.14 Distribution and Normality test of pH for red and white wine

Sulphates: Red wines appear to be slightly higher in the median, but each density plot appears right-skewed. Shapiro's test indicates a non-normal distribution.







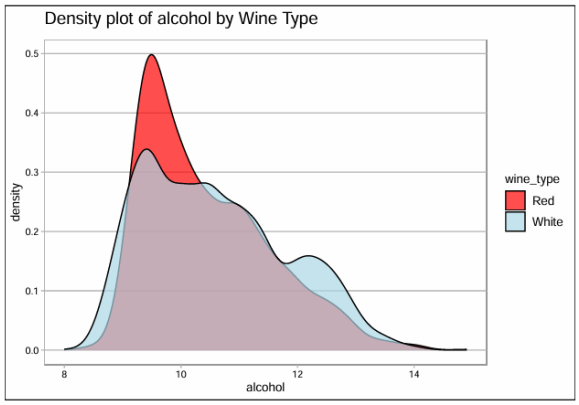


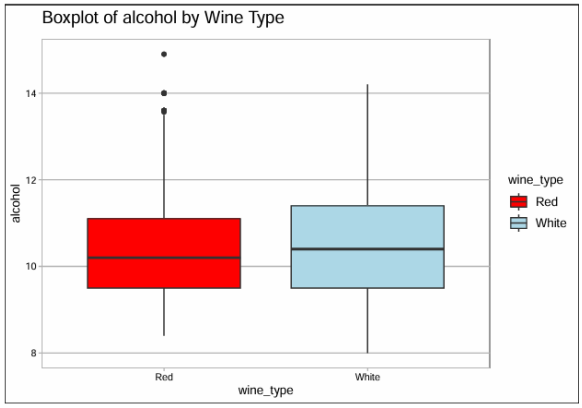

```
Shapiro-Wilk normality test

data:  column_data
W = 0.83024, p-value < 2.2e-16
```

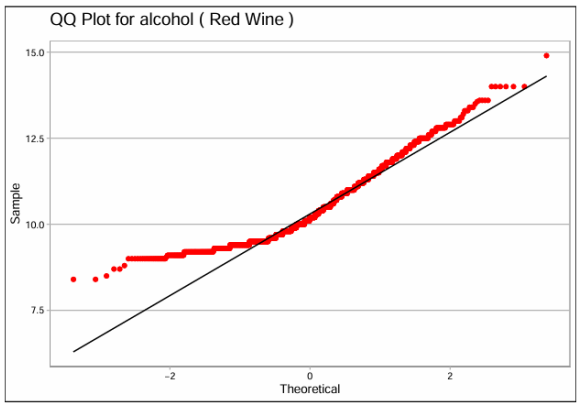
```
Shapiro-Wilk normality test

data:  column_data
W = 0.95589, p-value < 2.2e-16
```

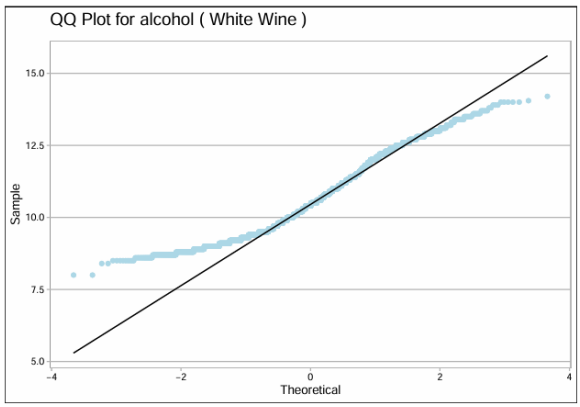
Figure 2.23 Figure 2.14 Distribution and Normality test of sulphates for red and white wine

Alcohol: White wine exhibits a slightly higher median alcohol than red wines, with very few outliers. They appear non-normal in the QQ plots.










```
Shapiro-Wilk normality test

data:  column_data
W = 0.9268, p-value < 2.2e-16
```

```
Shapiro-Wilk normality test

data:  column_data
W = 0.96191, p-value < 2.2e-16
```

Figure 2.24 Figure 2.14 Distribution and Normality test of alcohol for red and white wine

1.3 Correlation Analysis.

Understanding the data types will help in identifying the appropriate correlation technique to adopt.

- Continuous variables: All the physicochemical properties are continuous
- Nominal variables: Wine type is nominal
- Ordinal Variables: Quality is ordinal

1.3.1 Correlation Between Physicochemical Properties - Multiple Continuous Correlation

As established in the exploratory data analysis, all the physicochemical properties are non-normally distributed, so a Spearman correlation method is adopted (McAleer, 2022).

```
#continuous columns in the wine_data data frame
wine_data_continuous <- wine_data %>%
  select(-quality,-wine_type)

# correlation matrix: Multiple continuous variables
# as seen in the normality tests, all the continuous variables are non-normal so we use Spearman's
wine_data_continuous_cor_matrix <- round(cor(wine_data_continuous, method = "spearman"), digit=2)
wine_data_continuous_cor_matrix
```

```
# Visualizing the continuous correlation matrix
corrplot(wine_data_continuous_cor_matrix,
         method = "number",
         type = "upper",
         title = "Corrplot for Continous Variables (Wine data)",
         tl.col = "black",
         number.cex = 0.6,
         tl.cex = 0.6,
         cl.cex = 0.6,
         mar = c(1, 0, 2, 0))
```

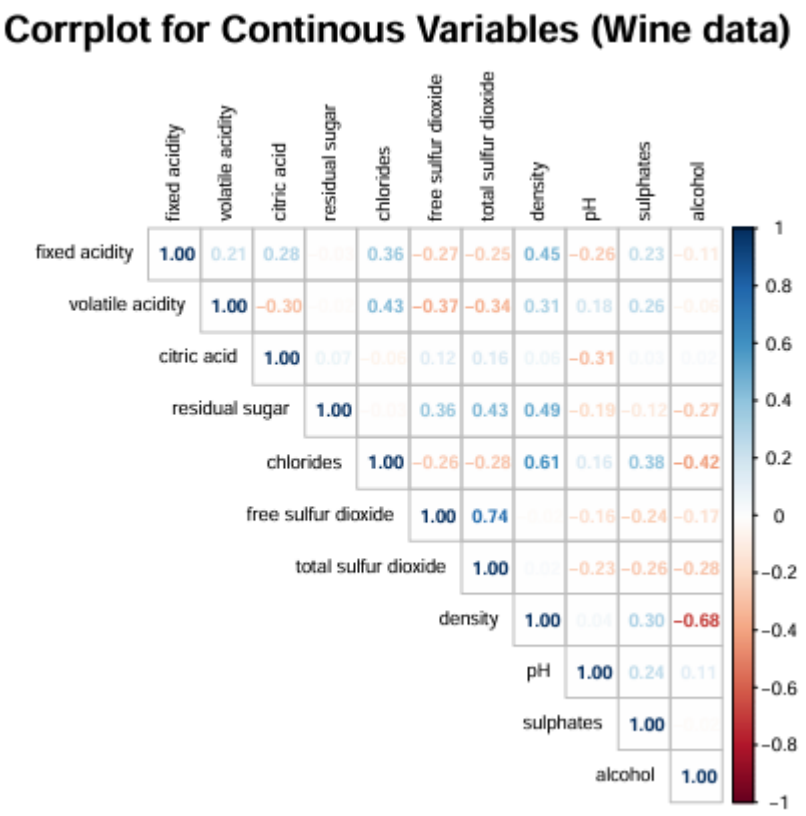


Figure 2.25 Correlation of physicochemical properties

1.3.2 Correlation Between Physicochemical Properties and Wine Type

Wine type is a nominal categorical variable therefore the correlation between wine type and the physicochemical properties can be measured with the Point-Biserial method (Kornbrot et al., 2005). First, the wine type was converted to binary, where red wine was encoded as 0 and white wine as 1.

```r
# Creating a copy of the wine_data df to introduce wine_type as numeric.
wine_data_copy <- wine_data %>%
  mutate(wine_type_numeric = ifelse(wine_type == "Red", 0, 1))
```

```r
# Using Point-biserial correlation with wine type
point_biserial_cor <- sapply(names(wine_data_continuous), function(var) {
  cor.test(wine_data_copy[[var]], wine_data_copy$wine_type_numeric, method = "pearson")$estimate
})

# Converting to a data frame for better readability
pb_cor_df <- data.frame(
  Variable = names(wine_data_continuous),
  pb_correlation = point_biserial_cor
)
print(pb_cor_df)
```

Figure 2.26 R code performing correlation analysis of physicochemical properties and wine type

The output is a data frame showing the correlation between wine type and each physicochemical property, which can be visualized for better interpretability.

```r
# Visualizing the point-biserial correlations between the continuous variables and wine type
ggplot(pb_cor_df, aes(x = reorder(Variable, pb_correlation),
                      y = pb_correlation,
                      fill = pb_correlation)) +
  geom_bar(stat = "identity") +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0) +
  labs(title = "Point-Biserial Correlation of Continuous Variables with Wine Type",
       x = "Continuous Variables") +
  theme_calc() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

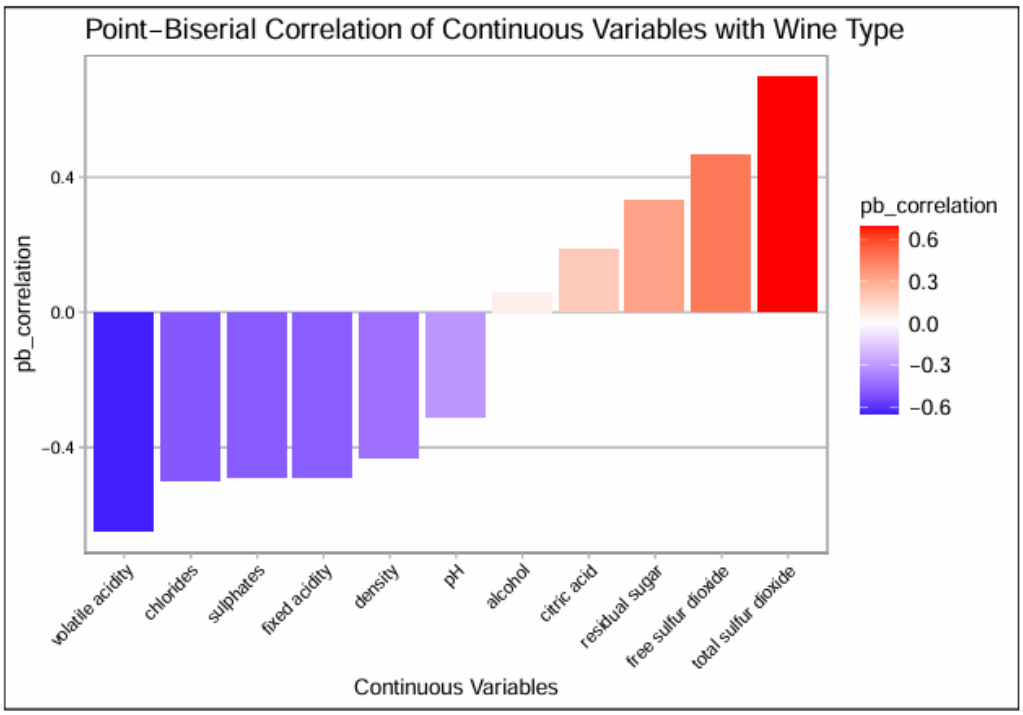


Figure 2.27 Chart showing the correlation of each physicochemical property with wine type

From the correlation chart, Volatile acidity, chlorides, free sulfur dioxide, and total sulfur dioxide show a high correlation with wine type.

1.3.3 Correlation Between Physicochemical Properties and Quality

Quality is an ordinal categorical variable hence spearman method is best to measure the correlation with continuous variables (Khamis, 2008).

```
# Spearman correlation of continuous variables with quality
spearman_cor <- sapply(names(wine_data_continuous), function(var) {
  cor.test(wine_data_copy[[var]], as.numeric(wine_data_copy$quality), method = "spearman")$estimate
})
```

```
# Converting results to a data frame
spearman_cor_df <- data.frame(
  Variable = names(wine_data_continuous),
  spearman_correlation = spearman_cor
)
print(spearman_cor_df)
```

Figure 2.28 R code performing correlation analysis of physicochemical properties and quality

The results can be visualized by using a bar chart.

```
# Visualizing the spearman correlation between the continuous variables and quality
ggplot(spearman_cor_df, aes(x = reorder(Variable, spearman_correlation),
                            y = spearman_correlation,
                            fill = spearman_correlation)) +
  geom_bar(stat = "identity") +
  scale_fill_gradient2(low = "green", mid = "white", high = "brown", midpoint = 0) +
  labs(title = "Spearman Correlation of Continuous Variables with quality",
       x = "Continuous Variables") +
  theme_calc() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

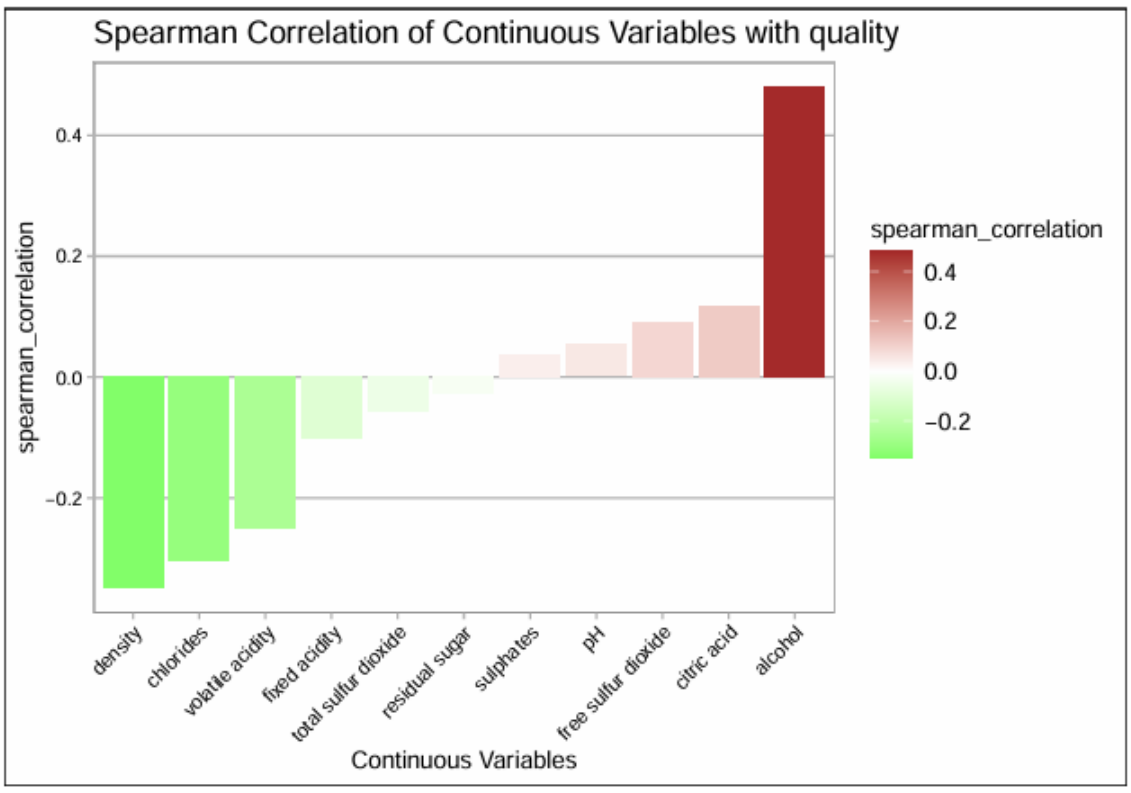


Figure 2.29 Chart showing the correlation of each physicochemical property with quality

1.4 Hypothesis Testing

1.4.1 Hypothesis 1 - Association Between Wine Type and Quality

Purpose: To determine whether wine type and quality are associated

- Null Hypothesis (H_0): There is no association between wine type and quality
- Alternative Hypothesis (H_1): Wine type and wine quality are associated

Test: Chi-Square Test of Independence.

Wine type and quality are both categorical variables so it is appropriate to use chi-square to test for independence to test for association between them. (McHugh, 2013). Quality with 7 levels has been reduced to three levels of 'Low', 'Medium', and 'High' to make the test less complex. Visualizing the number of wines in each quality group by wine type can help understand the distribution.

```
# Creating quality type to reduce the perform Chi-squared test
wine_data_copy <- wine_data_copy %>%
  mutate(quality_type = case_when(
    quality %in% c(3, 4) ~ "Low",
    quality %in% c(5,6) ~ "Medium",
    quality %in% c(7,8, 9) ~ "High"
  ))
wine_data_copy$quality_type  <- factor(wine_data_copy$quality_type ,
                                       levels = c("Low", "Medium", "High"),
                                       ordered = TRUE)
table(wine_data_copy$wine_type,wine_data_copy$quality_type)
```

Figure 2.30 R code to bin quality into 3 classes

```
# Plotting number of wines by Quality and type
ggplot(wine_data_copy,aes(x = quality_type, fill = wine_type)) +
  geom_bar() +
  labs(title = "Number of Wines by Quality by Type") +
  scale_fill_manual(values = c("Red" = "red", "White" = "lightblue")) +  theme_calc()
```

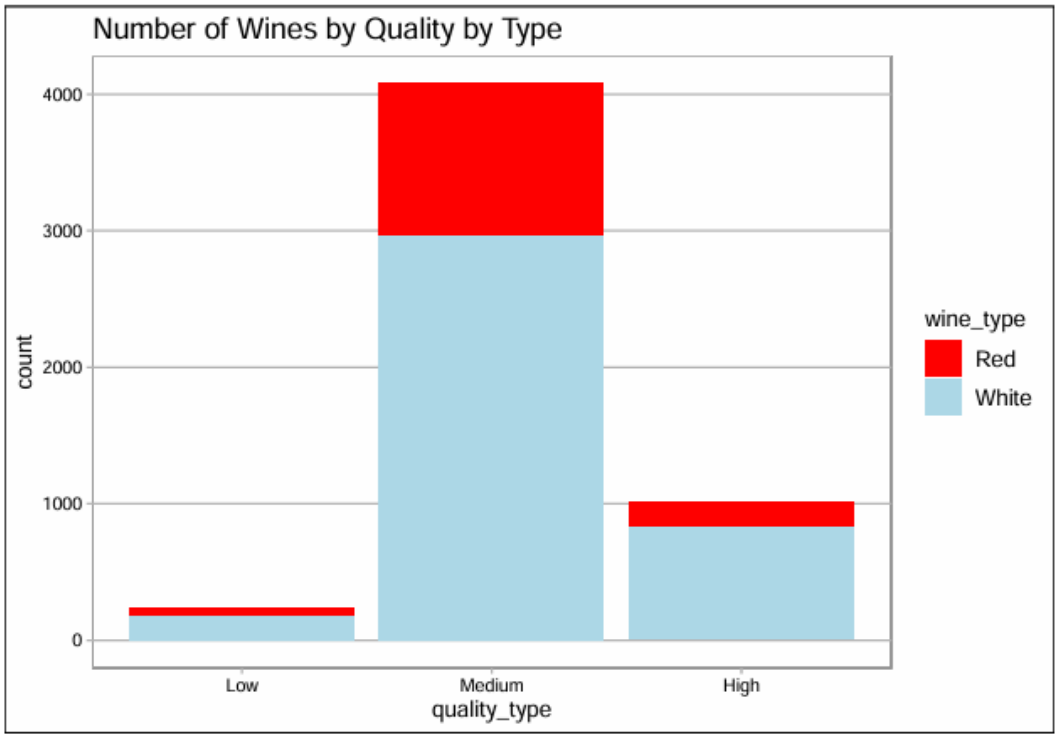


Figure 2.31 A stacked bar chart showing the number of wines in each quality group

```r
# Chi-squared test of independence.
chisq_test <- chisq.test(wine_data_copy$wine_type, wine_data_copy$quality_type)
chisq_test
```

```
##
##  Pearson's Chi-squared test
##
## data:  wine_data_copy$wine_type and wine_data_copy$quality_type
## X-squared = 35.017, df = 2, p-value = 2.49e-08
```

```r
chisq_test$residuals
```

```
##                          wine_data_copy$quality_type
## wine_data_copy$wine_type         Low     Medium       High
##                    Red    0.3494824  2.2017336 -4.5937106
##                    White -0.2047070 -1.2896507  2.6907351
```

Figure 2.32 Chi-squared test of independence for wine type and quality

The Null hypothesis is rejected because the p-value is less than 0.05, which is the significance threshold. This means there is a statistically significant association between quality and the type of wine. The residuals also indicate that there are fewer high-quality red wines than expected and there are slightly more medium-quality red wines than expected.

1.4.2 Hypothesis 2 – Difference in mean pH between red and white wines.

Purpose: To compare the mean pH between red and white wines.

- Null Hypothesis (H_0): The mean pH levels of red and white wine are equal.
- Alternative Hypothesis (H_1): The mean pH levels of red and white wine are not equal.

Test: Two-Sample t-test

Before a two-sample t-test can be carried out, the following assumptions must be satisfied:

- Independence: No repeated measures of pH for a single wine sample.
- Normality: pH for each wine type should follow a normal distribution (Bell Curve).
- Homogeneity of variance: The variance of pH should be similar in both wine types.

Independence Check: By nature of the data sets, each observation is independent.

Normality Check: The EDA carried out earlier established that all the physicochemical properties appear to be non-normal. Log transformations can be carried out on the variable to solve this.

```r
# Visualizing the distribution of log_pH based on wine type.
wine_data %>%
  mutate(log_pH = log(pH)) %>%
  ggplot(aes(x = wine_type, y = log_pH, fill = wine_type)) +
    geom_boxplot() +
    labs(title = paste("Boxplot of log_pH by Wine Type")) +
    scale_fill_manual(values = c("Red" = "red", "White" = "lightblue")) +
    theme_calc() +
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
```

Figure 2.33 R code to plot the distribution of log_pH for both wine types

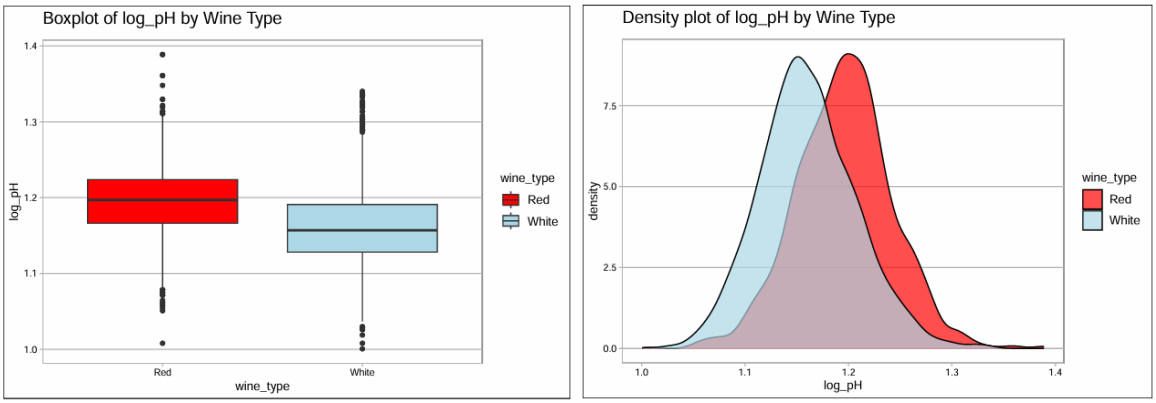


Figure 2.34 Distribution of log_pH

The boxplot shows the distribution of log_pH for both wine types, the median of red wine appears to be higher. The density plot appears to be in the shape of a bell curve, but further analysis is better.

```
# Shapiro-Wilk test for normality
wine_data %>%
  group_by(wine_type) %>%
  mutate(log_pH = log(pH)) %>%
  summarise(p_value = shapiro.test(log_pH)$p.value)
```

```
## # A tibble: 2 x 2
##   wine_type  p_value
##   <fct>        <dbl>
## 1 Red       2.83e- 4
## 2 White     2.21e-11
```

Figure 2.35 Shapiro-Wilk's normality test for log_pH

The p-value of log_pH for red and white wine is less than 0.05, which suggests the null hypothesis be rejected. But the density plot approximately resembles a bell curve.

```
# QQ plot check for normality
red_wine %>%
  mutate(log_pH = log(pH)) %>%
  ggplot(aes(sample = log_pH)) +
    stat_qq(color ='red') +
    stat_qq_line(color = "black") +
    labs(
      title = paste("QQ Plot for log_pH of red wine"),
      x = "Theoretical",
      y = "Sample") +
    theme_calc()+
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
```

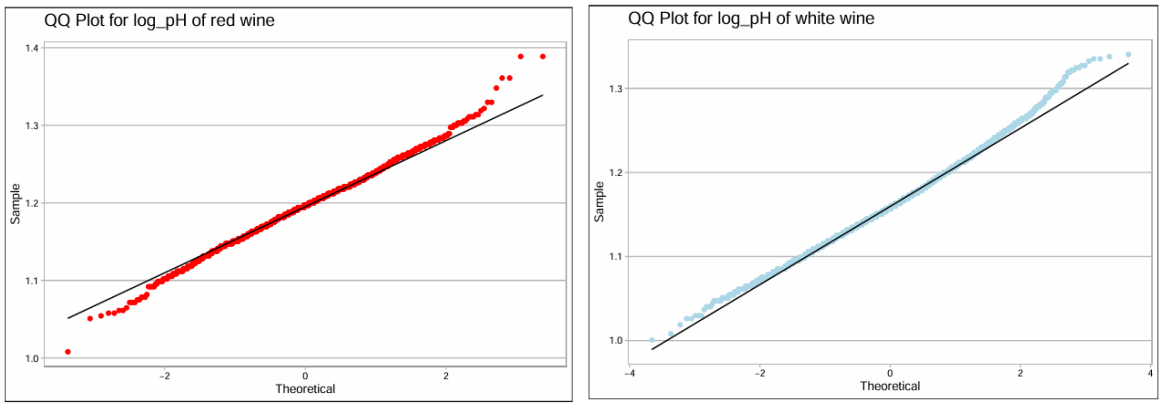


Figure 2.36 QQ-plot normality test for log_pH

From the QQ plot, it is safe to assume that log_pH is normally distributed for red and white wines. Hence passing the normality check.

Homogeneity Check: The Bartlett test can be used to test the homogeneity of variance in log_pH for both wine types(Arsham & Lovric, 2011).

```
# Test for homogeneity of variance
wine_data %>%
  mutate(log_pH = log(pH)) %>%
  bartlett.test(log_pH ~ wine_type, data = .)
```

```
 Bartlett test of homogeneity of variances

data:  log_pH by wine_type
Bartlett's K-squared = 0.089408, df = 1, p-value = 0.7649
```

Figure 2.37 R code and result for Barlett test for homogeneity in variance

The Null hypothesis of the Bartlett test is that the variance is equal for the two groups, since the p-value of 0.7649 is greater than 0.05, The null hypothesis is accepted and homogeneity in variance assumption has been passed. Now the test can be carried out.

Two-sample t-test

```
# Two sample t-test (two tailed)
wine_data %>%
  mutate(log_pH = log(pH)) %>%
  t.test(log_pH ~ wine_type, data = ., var.equal = TRUE)
```

```
 Two Sample t-test

data:  log_pH by wine_type
t = 23.813, df = 5318, p-value < 2.2e-16
sample estimates:
  mean in group Red mean in group White
           1.195791            1.160618
```

Figure 2.38 Two-sample t-test for log_pH

The result from the test shows $p < 0.05$, therefore the Null Hypothesis is rejected and the Alternative Hypothesis that the mean pH levels of red and white wine are not equal is accepted.

1.4.3 Hypothesis 3 – Difference In Chloride Between Red and White Wines

Purpose: Assess the difference in the distribution difference in chlorides for red and white wines

- Null Hypothesis (H_0): The distribution of chlorides is identical in red and white wines.
- Alternative Hypothesis (H_1): The distribution of chlorides is not identical in red and white wines.

Parametric Test? Before the parametric test can be carried out, the distribution of chlorides in the two groups tested must be normally distributed.

Normality Check: The EDA confirmed that chloride is not normally distributed for both red and white wines as shown in the QQ plot below.

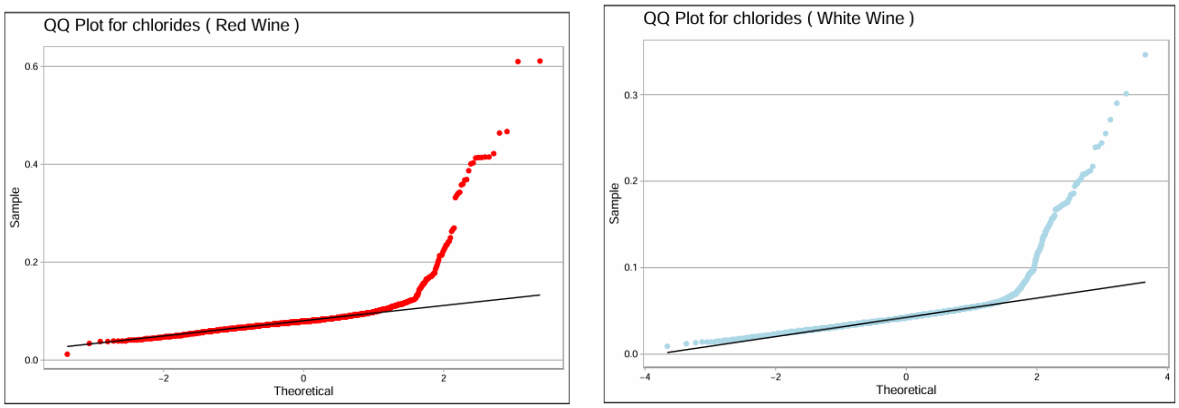


Figure 2.39 QQ-plot of chlorides for red and white wine

Transformations may be carried out to make distribution approximately normal. A function was created to carry out these transformations and return the QQ plot. Note similar functions were used for all transformations.

```r
# function to plot the log transformed QQ plot
log_qqplot <- function(data, column_name, color, wine_type) {
  qqplot <- ggplot(data, aes(sample = log(.data[[column_name]]))) +
    stat_qq(color = color) +
    stat_qq_line(color = "black") +
    labs(
      title = paste("QQ Plot for log", column_name, "(", wine_type, ")"),
      x = "Theoretical",
      y = "Sample"
    ) +
    theme_calc()+
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
  print(qqplot)
}
```

Figure 2.40 R-code creating a function to plot QQ-plot of log-transformed data.

1. **Log Transformation:** The transformation failed to make chlorides approximately normal

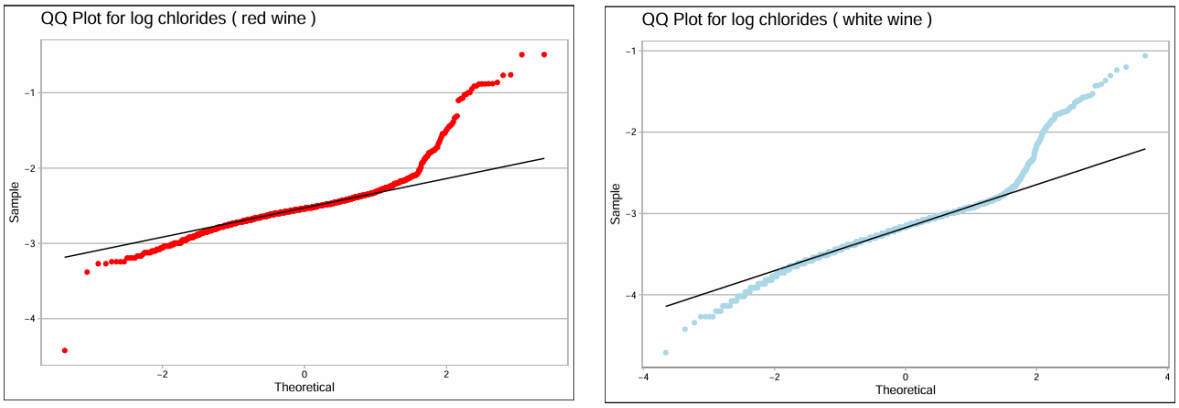


Figure 2.41 QQ-plot of log-transformed chlorides for red and white wine

2. **Square Root Transformation:** As shown in the QQ plot below, the transformation failed to make chlorides approximately normal.

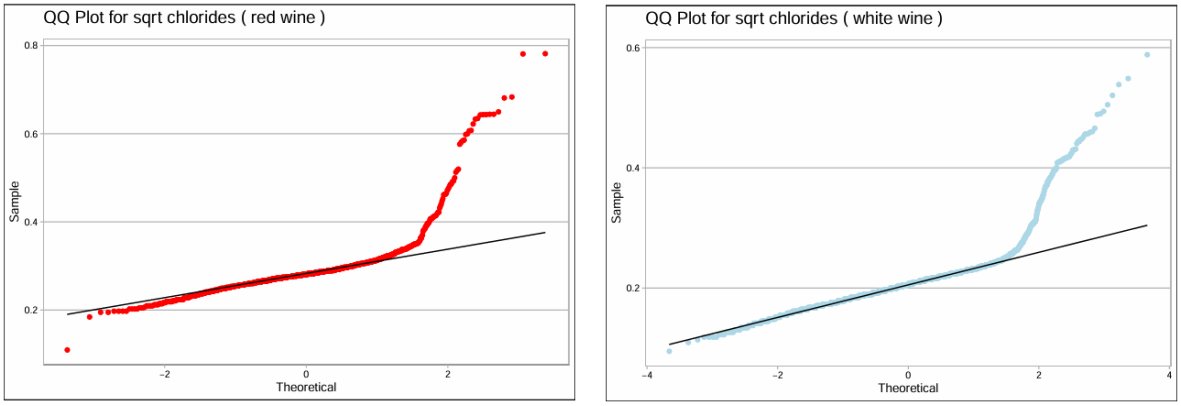


Figure 2.42 QQ-plot of square root transformed chlorides for red and white wine

3. **Cube Root Transformation:** The transformation also failed to make chlorides approximately normal.

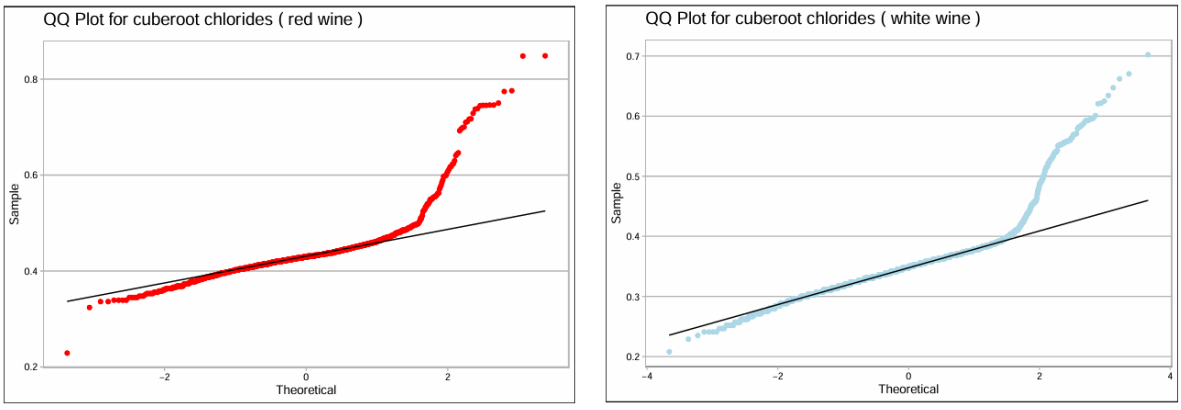


Figure 2.43 QQ-plot of cube root transformed chlorides for red and white wine

After all the transformations, it is safe to conclude that normality has been violated, hence a non-parametric test like the Mann-Whitney U Test can be considered (Nachar, 2008).

1.4.3.1 Test: Mann-Whitney U Test

This test assumes a similar shape in the distribution of chlorides for both wine types.

```
#plotting the distribution of chlorides for red  wine
  ggplot(red_wine, aes(x = chlorides)) +
    geom_histogram(alpha = 0.7, bins =10,fill = "red") +
    labs(title ="Histogram of chlorides (Red Wine)") +
    theme_calc() +
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
```

```
#plotting the distribution of chlorides for white wine
  ggplot(white_wine, aes(x = chlorides)) +
    geom_histogram(alpha = 0.7, bins =10, fill = "lightblue") +
    labs(title ="Histogram of chlorides (White Wine)") +
    theme_calc() +
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
```

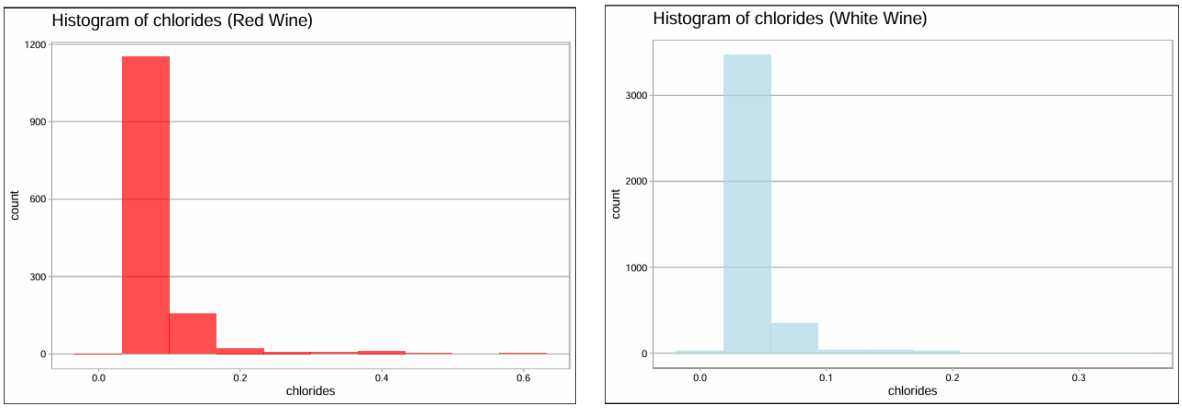


Figure 2.44 Histogram showing the distribution of chlorides in red and white wine

The Histogram of both red and white wine distribution of chlorides has a similar shape, therefore satisfying the assumptions for the test.

```
# Mann-Whitney U Test / Wilcox rank sum test
wilcox.test(chlorides ~ wine_type,data = wine_data)
```

```
 Wilcoxon rank sum test with continuity correction

data:  chlorides by wine_type
W = 5071770, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0
```

Figure 2.45 R code and output for the Mann-Whitney U test

P<0.05, the Null hypothesis is rejected: The distribution of chlorides differs significantly between the two wine types.

1.4.4 Hypothesis 4 – Difference in Alcohol Content by Wine Quality

Purpose: Compare median alcohol levels across wine quality (low, medium, high).

- Null Hypothesis (H_0): The median alcohol content is the same across all quality levels.
- Alternative Hypothesis (H_1): The median alcohol content differs across all quality levels.

Parametric Test? Before the parametric test can be carried out, the distribution of alcohol across the three quality levels must be normally distributed.

Normality Check: The P-Values obtained from the Shapiro-Wilks test suggest that alcohol is not normally distributed in the three quality groups.

```
# Normality check for alcohol for all quality types
byf.shapiro(alcohol ~ quality_type, data = wine_data_copy)
```

```
 Shapiro-Wilk normality tests

data:  alcohol by quality_type

           W   p-value
Low    0.9742 0.0002668 ***
Medium 0.9452 < 2.2e-16 ***
High   0.9870 8.491e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 2.46 Shapiro-Wilks normality test for alcohol in three quality types.

P < 0.05 in the three quality groups, hence the Null hypothesis of normal distribution is rejected.

1.4.4.1 Test: Kruskal-Wallis Test

The Kruskal-Wallis test is non-parametric and can be used to compare the median of two or more independent groups. Before it can be carried out, the following assumptions must be satisfied:

- Independence
- Homogeneity of variance
- Similar shape distribution.

The independence check has been satisfied because the observations aren't paired with each other.

Homogeneity of variance: The Bartlett test shows the p-value is greater than 0.05, so the null hypothesis of equal variance across the quality groups is accepted.

```
# Test for homogeneity of variance
bartlett.test(alcohol ~ quality_type, data = wine_data_copy)
```

```
 Bartlett test of homogeneity of variances

data:  alcohol by quality_type
Bartlett's K-squared = 5.4887, df = 2, p-value = 0.06429
```

Figure 2.47 Barlett test for homogeneity in variance of alcohol in quality type

Similar Shape Distribution: The boxplot shows the median alcohol content in high-quality wine is substantially higher than in low and medium-quality wines.

```
# Distribution of alcohol in across quality type
ggplot(data=wine_data_copy, aes(x = quality_type, y = alcohol, fill=quality_type)) +
  geom_boxplot() +
  labs(title = paste("Alcohol level for each Quality Type")) +
  scale_fill_manual(values = c("Low" = "lightgreen", "Medium" = "green", "High" = "darkgreen")) +
  theme_calc() +
  theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
```

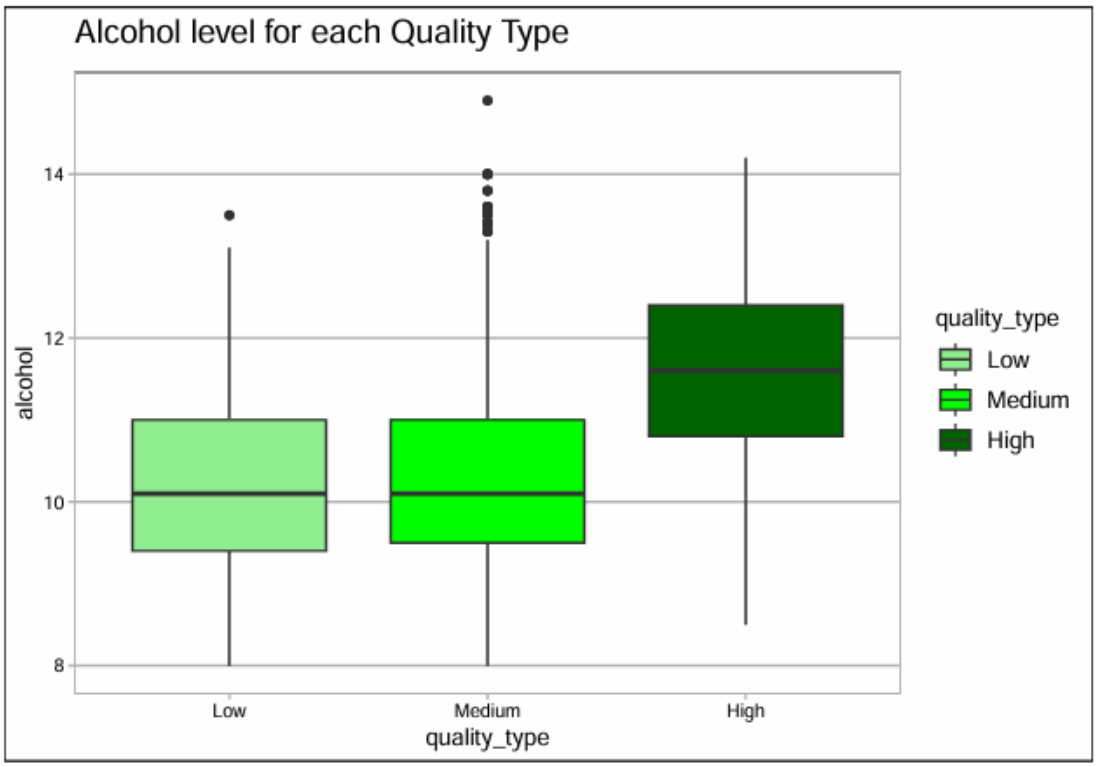


Figure 2.48 Box plot showing the distribution of alcohol for each quality type

```r
#plotting the distribution of Alcohol for low quality
wine_data_copy %>%
  filter(quality_type == 'Low') %>%
  ggplot(aes(x = alcohol)) +
    geom_density(fill = 'lightgreen') +
    labs(title ="Histogram of Alcohol for Low Quality") +
    theme_calc() +
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
```

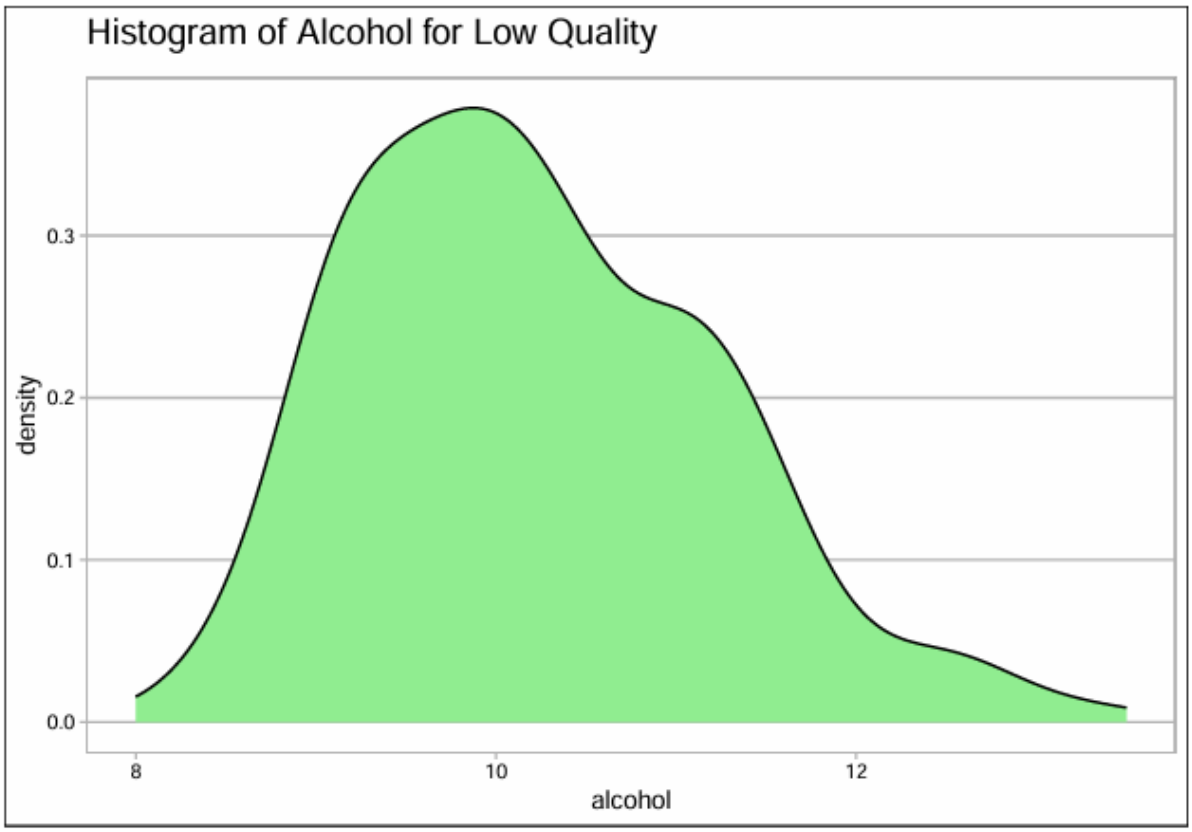


Figure 2.49 Density plot showing the distribution of alcohol in low-quality wines

Figure 2.49 shows that the distribution of alcohol content in low-quality wines is slightly right-skewed. Which means that more low-quality wines have less alcohol content.

Figure 2.50d shows the distribution of alcohol content in medium-quality wines is right skewed. It peaks at an alcohol content of roughly 9.5. It has a similar distribution to the low-quality wines.

```
#plotting the distribution of Alcohol for medium quality
wine_data_copy %>%
  filter(quality_type == 'Medium') %>%
  ggplot(aes(x = alcohol)) +
    geom_density(fill = 'green') +
    labs(title ="Histogram of Alcohol for Medium Quality") +
    theme_calc() +
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
```

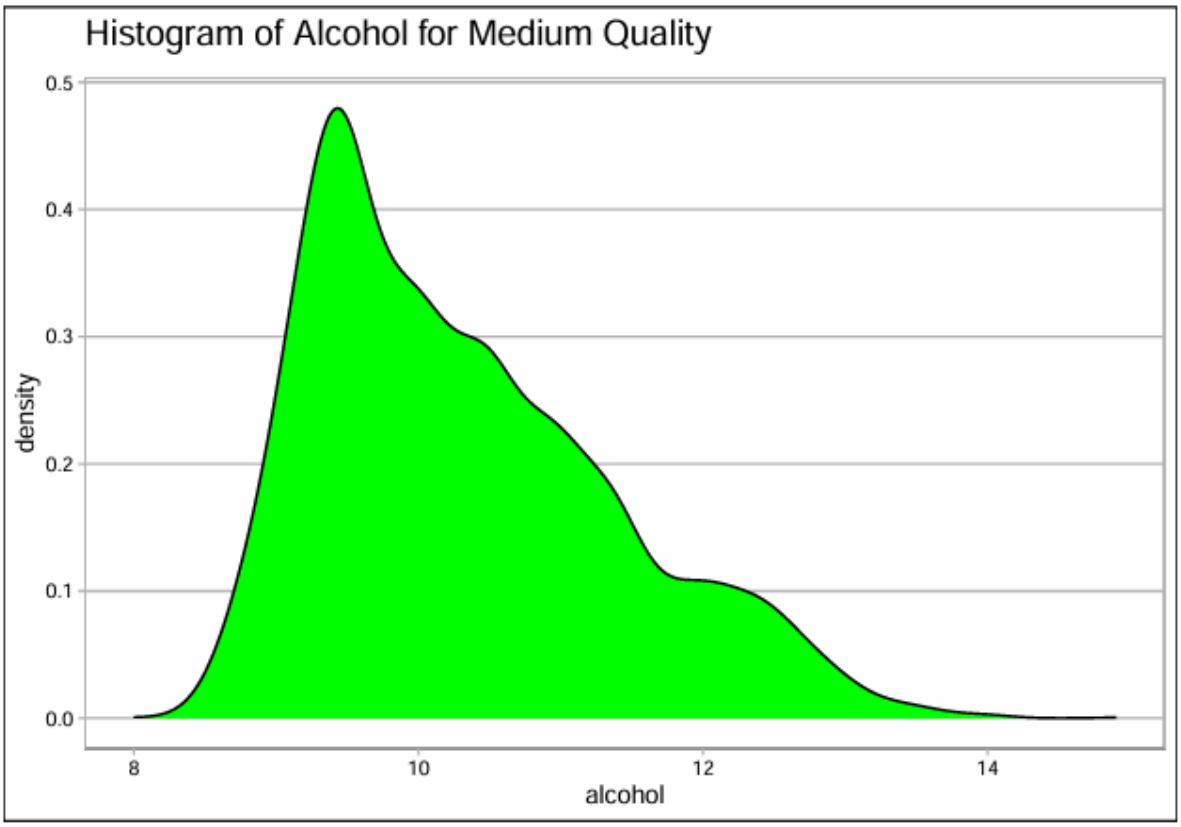


Figure 2.50 Density plot showing the distribution of alcohol in medium-quality wines

Figure 2.51 shows the distribution of alcohol content in high-quality wines is slightly left-skewed, suggesting more high-quality wines have higher alcohol content. This shape is different from the low and medium-quality wines which are right-skewed.

```
#plotting the distribution of Alcohol for medium quality
wine_data_copy %>%
  filter(quality_type == 'Medium') %>%
  ggplot(aes(x = alcohol)) +
    geom_density(fill = 'green') +
    labs(title ="Histogram of Alcohol for Medium Quality") +
    theme_calc() +
    theme(plot.title = element_text(size = 14, margin = margin(b = 10)))
```

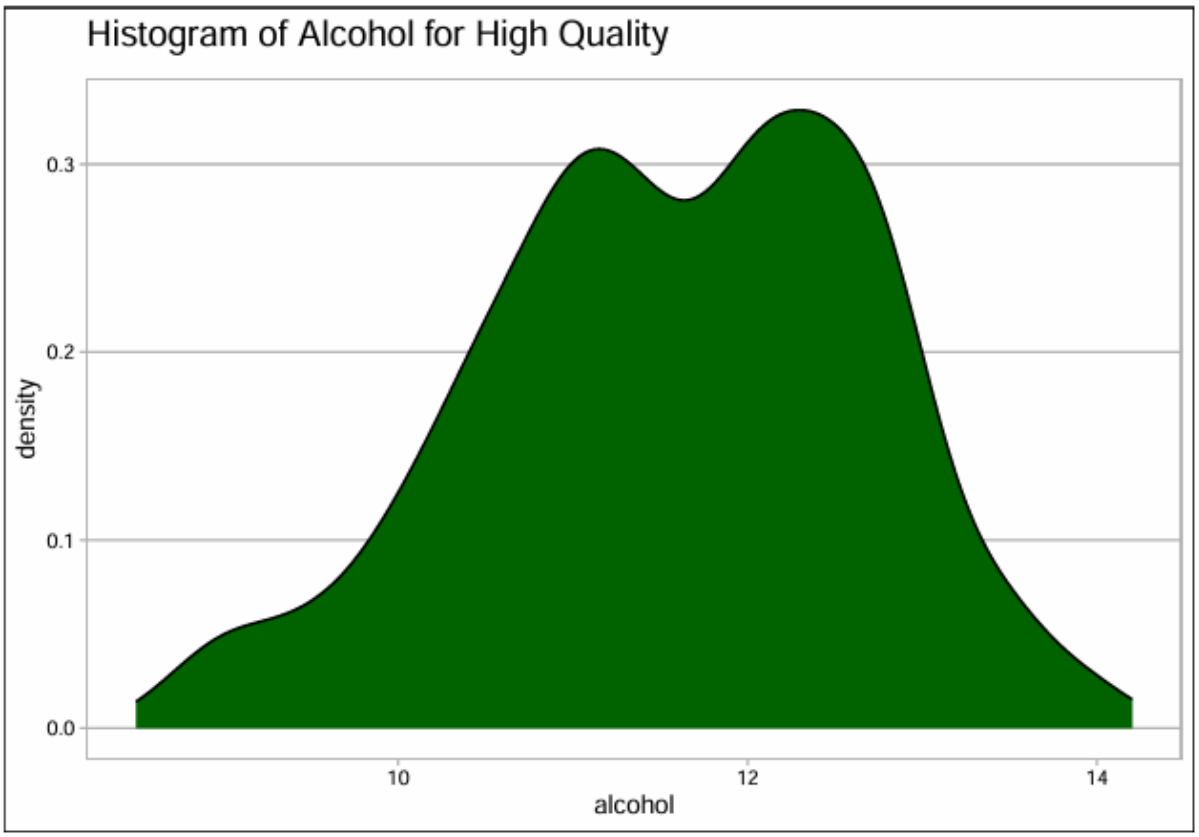


Figure 2.51 Density plot showing the distribution of alcohol in high-quality wines

Kruskal-Wallis Test

```
# Kruskal-Wallace test for non parametric models.
kruskal.test(alcohol ~ quality_type, data = wine_data_copy)
```

```
 Kruskal-Wallis rank sum test

data:  alcohol by quality_type
Kruskal-Wallis chi-squared = 838.84, df = 2, p-value < 2.2e-16
```

Figure 2.52 R code and output of Kruskal Wallis test

P < 0.05, The alternative hypothesis is accepted that median alcohol content differs across all quality levels. A post-hoc test can be carried out to find out which of the quality levels differ in alcohol content.

1.4.4.2 Post-Hoc Test: Dunn's Test

Dunn's test for pairwise multiple comparison is the most commonly used test when dealing with non-parametric samples (Zar, 1999).

```
# Dunns test
dunnTest(wine_data_copy$alcohol, wine_data_copy$quality_type, method = 'bonferroni')
```

```
  p-values adjusted with the Bonferroni method.

     Comparison         Z       P.unadj         P.adj
1    High - Low 14.819268  1.099756e-49  3.299269e-49
2 High - Medium 28.692487 4.734457e-181 1.420337e-180
3  Low - Medium -0.935171  3.497002e-01  1.000000e+00
```

Figure 2.53 code and output of Dunn's posthoc test

The p-value for high-low and high-medium quality wines is less than 0.05, meaning there is a statistically significant difference in their medium alcohol content. However low-medium quality wines have a p-value greater than 0.05, which suggests that there is no difference in the median alcohol content of these two groups.

1.5 Regression

1.5.1 Model 1 - Predicting Wine Quality

Purpose: Predict wine quality (high or low) based on physicochemical properties and wine type.

Model: Logistic Regression

Justification: To carry out logistic regression, the dependent variable must be categorical with only two classes, and the independent variable can be either categorical or numerical (Kassambara, 2018).

Model Description

- Dependent Variable: Quality (0 = low, 1 = high) – Categorical variable.
- Independent Variables: Physicochemical properties and wine type.
- Method: Backward stepwise.

```
# preparing data for logistic regression
# changing the quality to binary for logistic regression
wine_data_logistic <- wine_data %>%
  mutate(quality_binary = case_when(
    quality %in% c(3, 4,5,6) ~ 0,
    quality %in% c(7,8, 9) ~ 1
  ))
```

Figure 2.54 Code converting wine quality to binary

Quality which has 7 levels is first converted to binary, to make it suitable for logistic regression. All physicochemical properties and wine time are used as predictors following the backward stepwise method, then the model is tuned to leave only the most significant predictors.

```r
# Quailty prediction model 1
quality_prediction_1 <- glm(
  quality_binary ~ wine_type + `fixed acidity` + `volatile acidity` +
  `citric acid`+ `residual sugar` + chlorides+ `free sulfur dioxide` +
  `total sulfur dioxide`+ sulphates + density + pH + alcohol,
  data = wine_data_logistic,family = "binomial"
)

summary(quality_prediction_1)
```

```
Call:
glm(formula = quality_binary ~ wine_type + `fixed acidity` +
    `volatile acidity` + `citric acid` + `residual sugar` + chlorides +
    `free sulfur dioxide` + `total sulfur dioxide` + sulphates +
    density + pH + alcohol, family = "binomial", data = wine_data_logistic)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)           4.207e+02  7.391e+01   5.692 1.26e-08 ***
wine_typeWhite       -5.508e-01  2.718e-01  -2.026  0.04274 *
`fixed acidity`       5.326e-01  7.580e-02   7.027 2.12e-12 ***
`volatile acidity`   -3.044e+00  4.294e-01  -7.088 1.36e-12 ***
`citric acid`         3.541e-01  3.922e-01   0.903  0.36662
`residual sugar`      2.033e-01  2.972e-02   6.839 7.96e-12 ***
chlorides            -8.481e+00  2.797e+00  -3.033  0.00243 **
`free sulfur dioxide`  1.695e-02  3.399e-03   4.986 6.18e-07 ***
`total sulfur dioxide` -6.252e-03  1.553e-03  -4.026 5.67e-05 ***
sulphates             2.537e+00  3.266e-01   7.767 8.02e-15 ***
density              -4.451e+02  7.492e+01  -5.940 2.85e-09 ***
pH                    3.305e+00  4.100e-01   8.062 7.52e-16 ***
alcohol               4.764e-01  9.045e-02   5.267 1.39e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5168.2  on 5319  degrees of freedom
Residual deviance: 3937.0  on 5307  degrees of freedom
AIC: 3963

Number of Fisher Scoring iterations: 6
```

Figure 2.55 code and output for the first logistic regression model - quality_prediction_1

Quality_prediction_1 shows that citric acid is not an important predictor, so it will be removed and further fine-tuned to improve model performance.

```r
# Quality prediction model 2
quality_prediction_2 <- glm(
  quality_binary ~ wine_type + `fixed acidity` + `volatile acidity` +
  `residual sugar` + chlorides+ `free sulfur dioxide` +
  `total sulfur dioxide`+ sulphates + density + pH + alcohol,
  data = wine_data_logistic,family = "binomial"
)
```

Figure 2.56 code for logistic regression model - quality_prediction_2

```r
# Important features for predicting quality
quality_feature_importance <- varImp(quality_prediction_2, scale=False)
quality_feature_importance <- quality_feature_importance %>% arrange(desc(Overall))
quality_feature_importance
```

```
##                         Overall
## pH                     8.026870
## `volatile acidity`     7.859567
## sulphates              7.800256
## `fixed acidity`        7.388163
## `residual sugar`       6.821217
## density                5.894072
## alcohol                5.442944
## `free sulfur dioxide`  4.944129
## `total sulfur dioxide` 3.952710
## chlorides              2.961765
## wine_typeWhite         1.974587
```

Figure 2.57 Feature importance for quality_prediction_2

The most important features for predicting quality are pH, volatile acidity, sulfates, residual sugar, density, and alcohol.

1.5.1.1 Assumptions

Multicollinearity: VIF values of almost all of the predictors exceed 5, which is a problematic indicator of collinearity. This model has failed the assumption due to most of the VIF values exceeding 5.

```r
# Multicolinearity check
vif(quality_prediction_2)
```

```
            wine_type        `fixed acidity`     `volatile acidity`
             7.084829              6.289306              1.576465
      `residual sugar`             chlorides  `free sulfur dioxide`
             8.692678              2.175781              2.154992
`total sulfur dioxide`             sulphates               density
             3.695894              1.629069             27.810807
                   pH               alcohol
             2.739525              6.122816
```

Figure 2.58 VIF scores - multicollinearity check for quality_prediction_2

Quality_prediction_4 has been refitted to make sure there is no multicollinearity between the predictors.

```
# Quality prediction model 4
quality_prediction_4 <- glm(
  quality_binary ~ `volatile acidity` + wine_type +
  `residual sugar` + chlorides+ `free sulfur dioxide`+ sulphates + pH + alcohol,
  data = wine_data_logistic,family = "binomial"
)

summary(quality_prediction_4)
```

```
Call:
glm(formula = quality_binary ~ `volatile acidity` + wine_type +
    `residual sugar` + chlorides + `free sulfur dioxide` + sulphates +
    pH + alcohol, family = "binomial", data = wine_data_logistic)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -14.263451   1.066681 -13.372  < 2e-16 ***
`volatile acidity`    -3.905121   0.399006  -9.787  < 2e-16 ***
wine_typeWhite        -0.508452   0.177197  -2.869 0.004112 **
`residual sugar`       0.028225   0.011058   2.552 0.010699 *
chlorides            -11.213856   2.855071  -3.928 8.58e-05 ***
`free sulfur dioxide`  0.007837   0.002615   2.997 0.002725 **
sulphates              1.903407   0.303562   6.270 3.60e-10 ***
pH                     0.883657   0.258802   3.414 0.000639 ***
alcohol                0.983468   0.041382  23.766  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)
```

Figure 2.59 code and output for the logistic regression model - quality_prediction_4

Multicollinearity check: The VIF of all predictors falls between 1 and 3, which signifies there is no multicollinearity between the predictors.

```
# Multicolinearity check
vif(quality_prediction_4)
```

```
`volatile acidity`             wine_type      `residual sugar`
          1.533879              2.977820              1.274633
         chlorides `free sulfur dioxide`             sulphates
          2.100503              1.311159              1.399575
                pH               alcohol
          1.109331              1.335072
```

Figure 2.60 Multicollinearity check for quality_prediction_4

Linearity Check: For a model to be accepted, the independent and dependent variables should have a linear relationship. To achieve this, the logit(pi) is calculated.

```
# Calculating pi
probs_quality <- predict(quality_prediction_4, data=wine_data_logistic,type="response")
wine_data_logistic$probs_quality <- probs_quality
```

```
# Calulating logit(pi)
wine_data_logistic$logits_quality <- log(probs_quality/(1-probs_quality))
```

```
# scatter plot for linearity check
pairs(wine_data_logistic[,c(16,2,9,11)], lower.panel = NULL,
      upper.panel = panel.smooth, pch = 19,cex = 0.2)
```

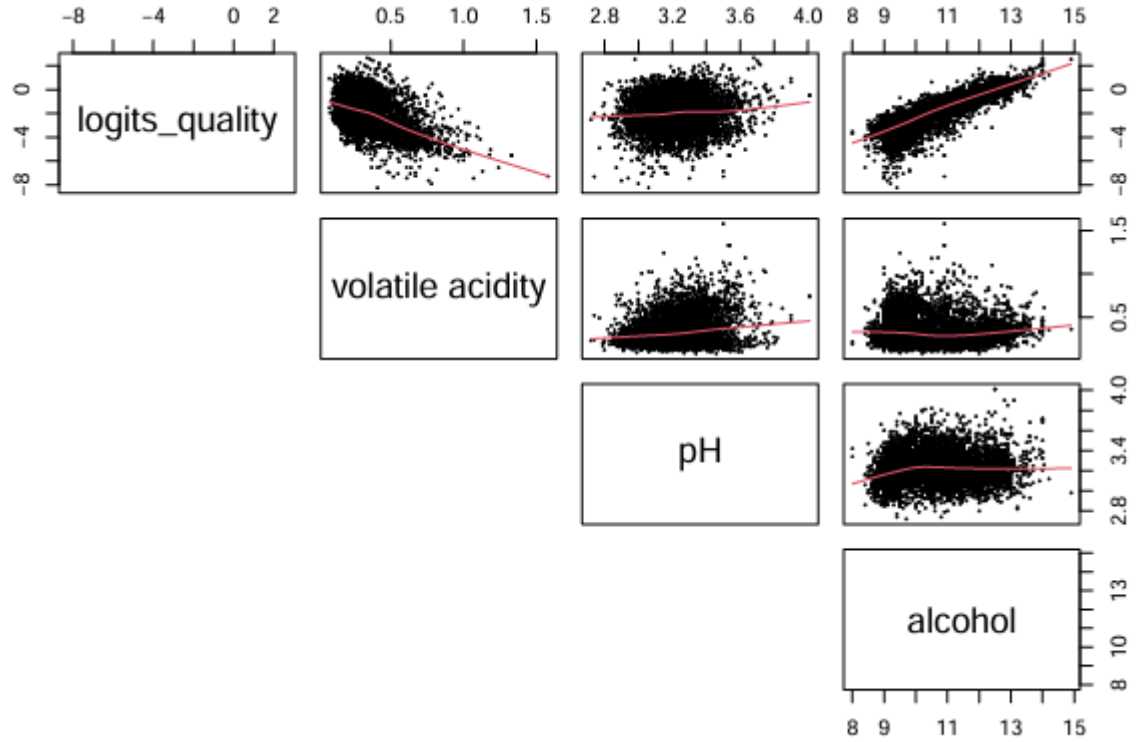


Figure 2.61 Scatterplots showing the linear relationships between logits_quality and predictors

The Logits appear to have an approximately linear relationship with volatile acidity, pH, and alcohol.

Influential Values: Cook's distance can be used to examine the most extreme values in the dataset. The visualization below shows that observations 246, 935, and 5187, might potentially contain outliers.

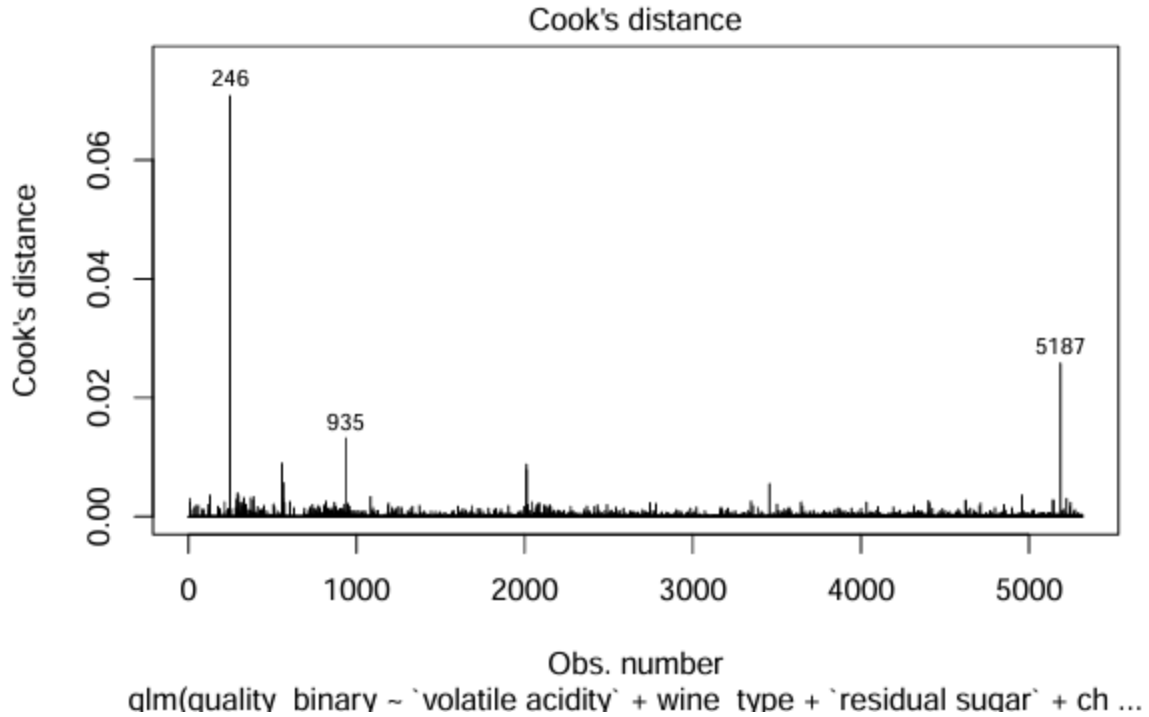


Figure 2.62 Cook's distance to check for influential values

1.5.1.2 Result

Based on the **quality_prediction_4** logistic regression model, this equation can be used to predict the quality of wines.

$$\ln\left(\frac{\pi}{1-\pi}\right) = \ln(odds) = Logit(\pi)$$
$$= -14.26 - 3.9(volatile\ acidity) - 0.5(wine\ type_{white})$$
$$+ 0.03(residual\ sugar) - 11.21(chlorides) + 0.008(free\ sulfur\ dioxide)$$
$$+ 1.9(sulphates) + 0.88(pH) + 0.98(alcohol)$$

1.5.2 Model 2 - Predicting Wine Type

Purpose: Classify wine type(red or white) based on physicochemical properties.

Model: Logistic Regression

Model Description

- Dependent Variable: Wine Type (0 = Red, 1 = White) – Nominal categorical variables.
- Independent Variables: Physicochemical properties – Continous variables.
- Method: Backward stepwise.

```
# data for prediction (removing columns observed with substantial outliers)
wine_data_logistic_2 <- wine_data %>%
  mutate(wine_type_numeric = ifelse(wine_type == "Red", 0, 1)) %>%
  slice(-c(3654,919,929))
```

Figure 2.63 code to remove observations with excessive outliers

Wine type was first converted to binary, then the predictors were fit into the model using the backward stepwise method.

```
# Second prediction model for wine type.
type_prediction_2 <- glm(
  wine_type_numeric ~ `volatile acidity` +
  `residual sugar`+ `total sulfur dioxide`+ density + alcohol,
  data = wine_data_logistic_2,family = "binomial"
)

summary(type_prediction_2)
```

```
Call:
glm(formula = wine_type_numeric ~ `volatile acidity` + `residual sugar` +
    `total sulfur dioxide` + density + alcohol, family = "binomial",
    data = wine_data_logistic_2)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)           2.094e+03  1.287e+02  16.265   <2e-16 ***
`volatile acidity`   -7.555e+00  8.880e-01  -8.508   <2e-16 ***
`residual sugar`      8.131e-01  6.521e-02  12.468   <2e-16 ***
`total sulfur dioxide` 4.911e-02  4.288e-03  11.454   <2e-16 ***
density              -2.086e+03  1.279e+02 -16.311   <2e-16 ***
alcohol              -2.129e+00  2.117e-01 -10.057   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6040.1  on 5316  degrees of freedom
Residual deviance:  372.4  on 5311  degrees of freedom
AIC: 384.4

Number of Fisher Scoring iterations: 9
```

Figure 2.64 Code and output for logistic regression model - type_prediction_2

```
# Most important variables when predicting wine type
type_feature_importance <- varImp(type_prediction_2, scale=False)
type_feature_importance <- type_feature_importance %>% arrange(desc(Overall))
type_feature_importance
```

```
                       Overall
density               16.310806
`residual sugar`      12.467764
`total sulfur dioxide` 11.454406
alcohol               10.057075
`volatile acidity`     8.508301
```

Figure 2.65 Important features in predicting wine type

1.5.2.1 Assumptions

Multicollinearity check: The VIF of all predictors is less than 5, therefore there is little to no collinearity between the predictors and the dependent variable.

```
# Multicolinearity check
vif(type_prediction_2)
```

```
`volatile acidity`       `residual sugar` `total sulfur dioxide`
          1.013125                2.041409                1.220932
           density                 alcohol
          4.334865                2.750990
```

Figure 2.66 Multicollinearity check

Linearity check: Logits have an approximately linear relationship with density, residual sugar, and alcohol.

```
# Calculating pi values
probs_type <- predict(type_prediction_2, data=wine_data_logistic_2,type="response")
wine_data_logistic_2$probs_type <- probs_type
```

```
# calculating (logit(pi))
wine_data_logistic_2$logits_type <- log(probs_type/(1-probs_type))
```

```
# Scatter plot to confirm linearity
pairs(wine_data_logistic_2[,c(16,8,4,11)], lower.panel = NULL,
      upper.panel = panel.smooth, pch = 19,cex = 0.2)
```

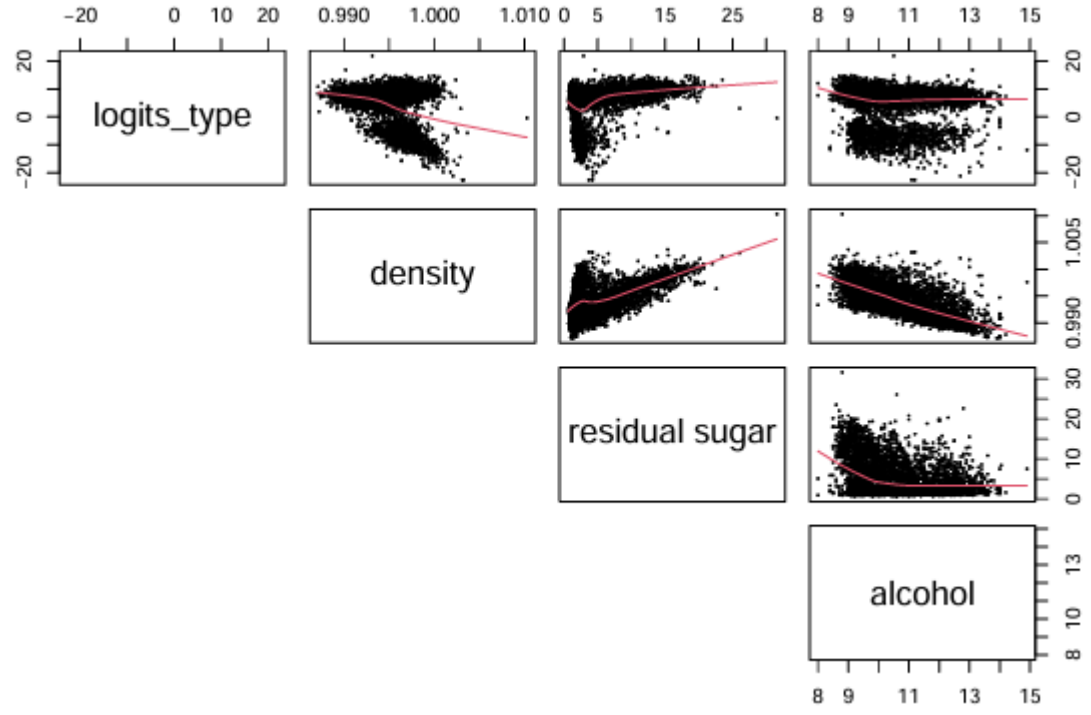


Figure 2.67 scatter plot showing linearity of logits_wine with the predictors

Influential Values: observation 920,2752 and 4516, have some influential values worth looking out for.

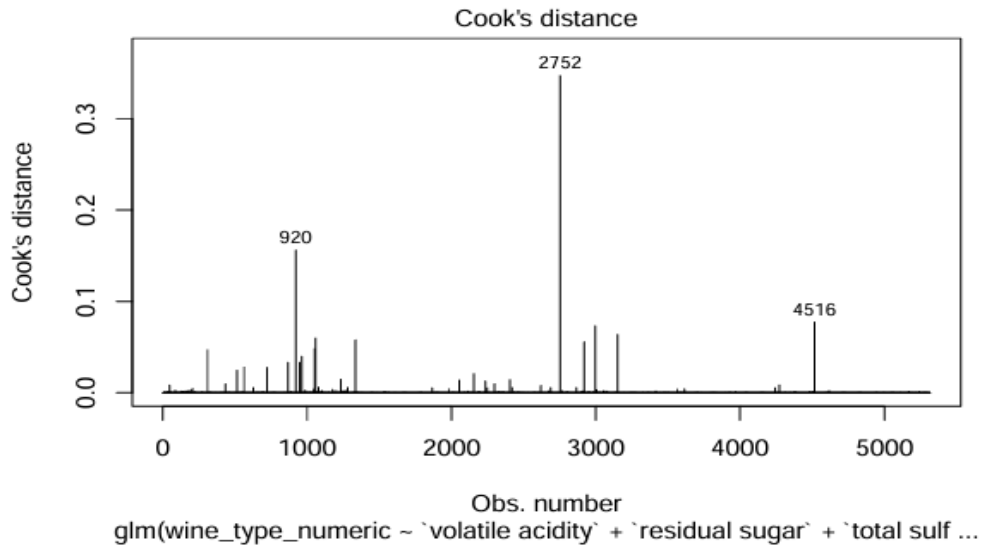


Figure 2.68 Cook's distance showing the top 3 influential values

1.5.2.2 Result

Based on the **type_prediction_2** logistic regression model, this equation can be used to classify the type of wines.

$$\ln\left(\frac{\pi}{1-\pi}\right) = \ln(odds) = Logit(\pi)$$
$$= 2094 - 7.56(volatile\ acidity) + 0.81(residual\ sugar)$$
$$+ 0.049(total\ sulfur\ dioxide) - 2086(density) - 2.13(alcohol)$$

1.5.3 Model 3 - Predicting the Density of Wine

Purpose: Wine density based on physicochemical properties.

Model: Multiple Linear Regression

Justification: To perform a linear regression, the independent and dependent variables must be continuous or discrete.

Model Description

- Dependent Variable: Density – Continuous variable.
- Independent Variables: Physicochemical properties – Continuous variables.

```r
# Visualizing the relationship between density and other properties
corrplot(wine_data_continuous_cor_matrix,
         method = "circle",
         type = "upper",
         title = "Relationship Between the Physiochemical Properties",
         tl.col = "black",
         number.cex = 0.6,
         tl.cex = 0.6,
         cl.cex = 0.6,
         mar = c(1, 0, 2, 0))
```

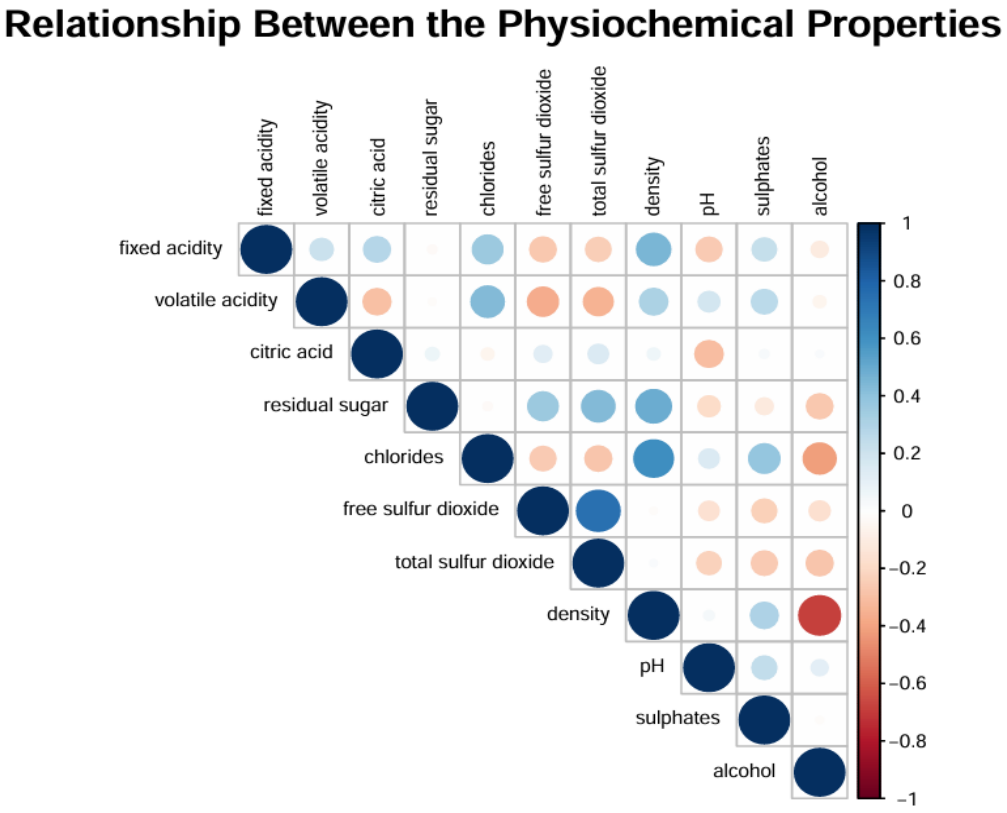


Figure 2.69 Correlation plot showing the relationship between the physiochemical properties

From Figure 2.69, density has a good correlation with alcohol, fixed acidity, volatile acidity, and residual sugar. These will be the predictors for the model.

```r
#Multiple Linear regression model
density_prediction <- lm(density ~ alcohol +`residual sugar` +
                         `fixed acidity`+ `volatile acidity`,
                         wine_no_outliers)
summary(density_prediction)
```

```
Call:
lm(formula = density ~ alcohol + `residual sugar` + `fixed acidity` +
    `volatile acidity`, data = wine_no_outliers)

Residuals:
       Min         1Q     Median         3Q        Max
-0.0042204 -0.0007157 -0.0000925  0.0006164  0.0055076
Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)         9.976e-01  2.089e-04 4776.57   <2e-16 ***
alcohol            -1.225e-03  1.438e-05  -85.17   <2e-16 ***
`residual sugar`    2.860e-04  4.013e-06   71.26   <2e-16 ***
`fixed acidity`     9.355e-04  1.523e-05   61.44   <2e-16 ***
`volatile acidity`  4.826e-03  1.099e-04   43.91   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.001107 on 4884 degrees of freedom
Multiple R-squared:  0.8428, Adjusted R-squared:  0.8427
F-statistic:  6548 on 4 and 4884 DF,  p-value: < 2.2e-16
```

Figure 2.70 Code and output for the multiple linear regression predicting density

1.5.3.1 Assumptions

Residual Independence: The correlation between the residuals is approximately 0, therefore the residuals are independent.

```r
# Residual Independence
plot(density_prediction,1)
```

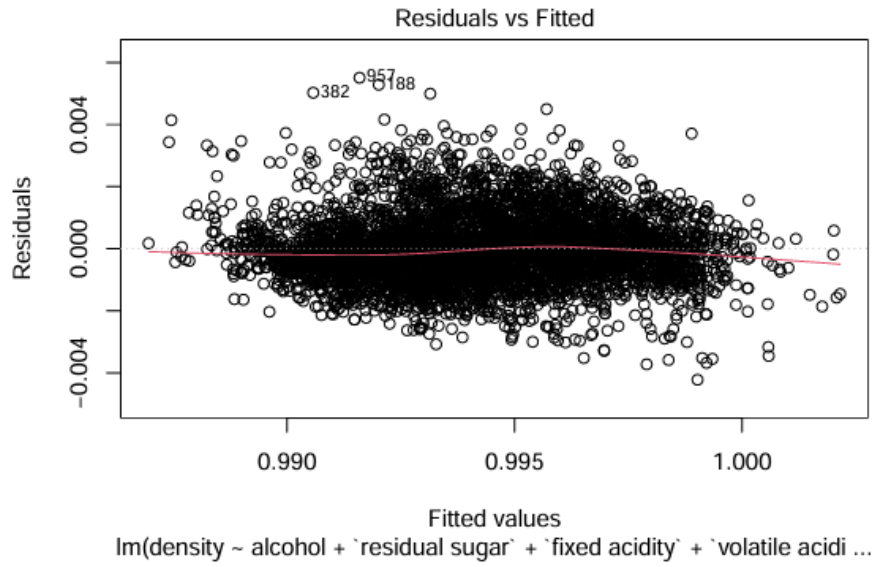
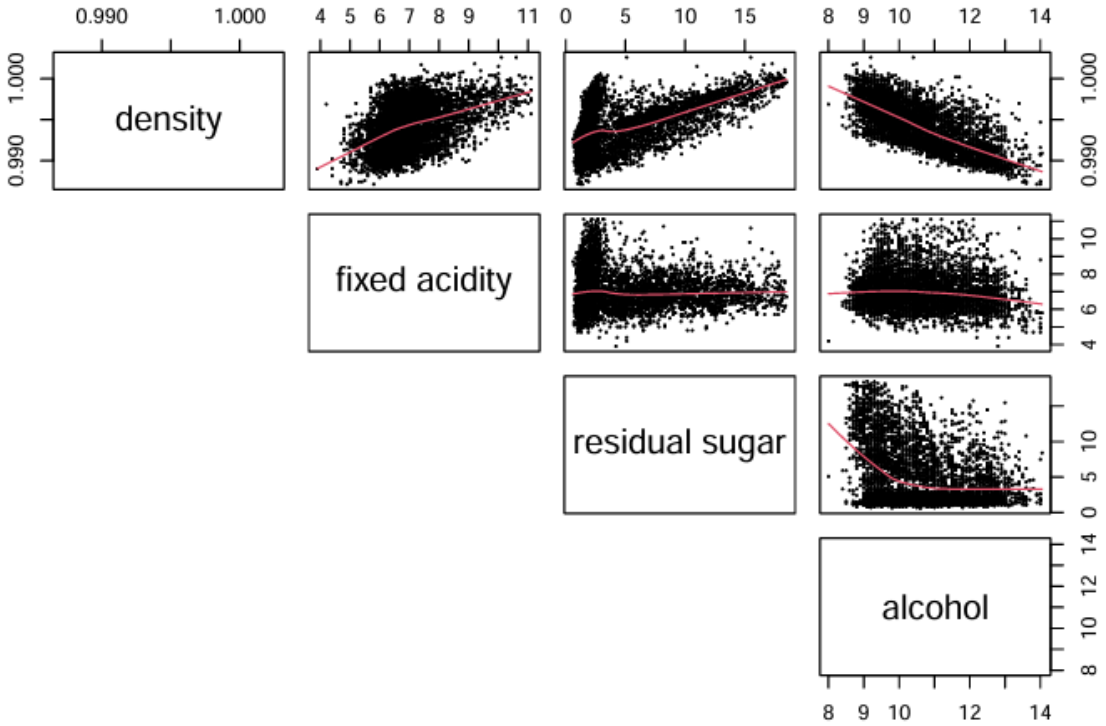


Figure 2.71 Plot showing residual independence

Linearity check: All the independent variables have a linear relationship with density (Dependent variable)

```
# Linearity between the IVs and DV
pairs(wine_no_outliers[,c(8,1,4,11)], lower.panel = NULL,
      upper.panel = panel.smooth,pch = 19,cex = 0.2)
```



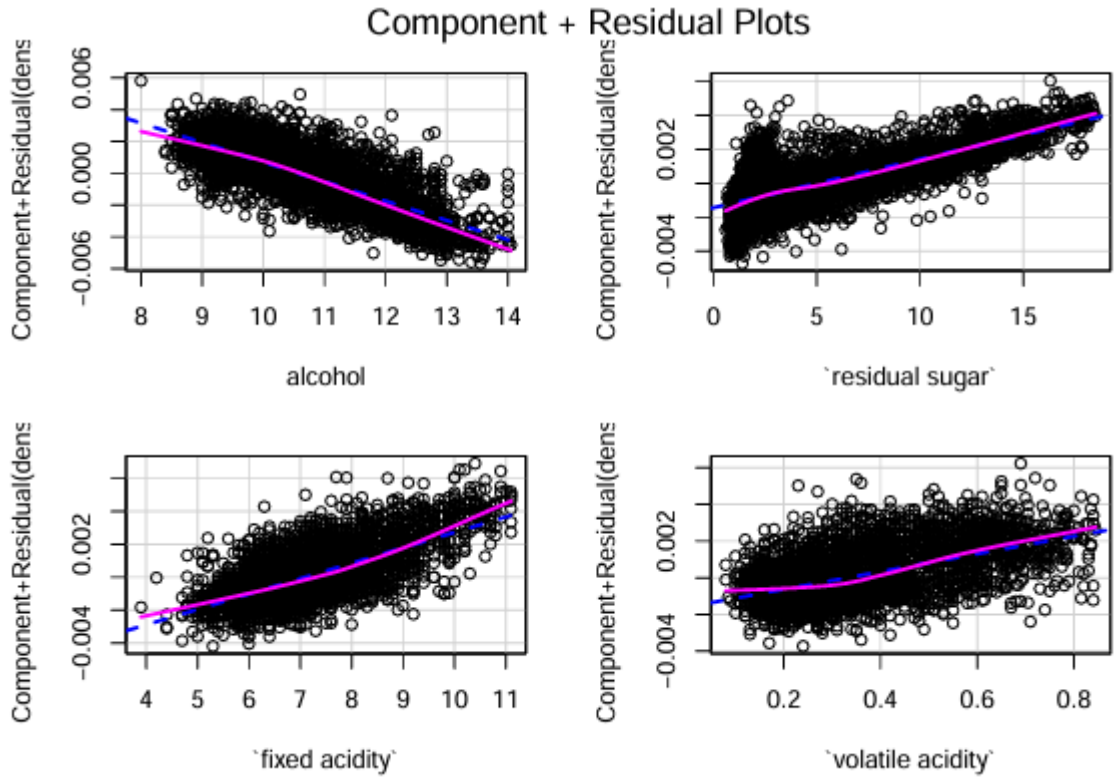


Figure 2.72Scatter plots showing linearity of residuals

Normality of residuals: The residuals appear approximately normally distributed.

```
# Normality of residuals
plot(density_prediction,2)
```

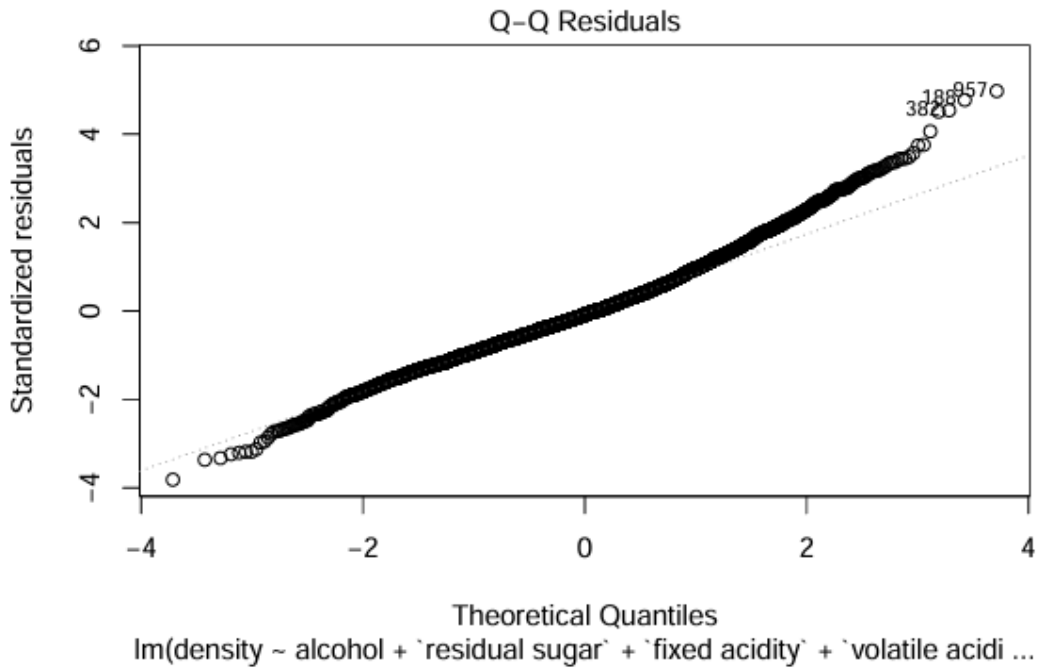


Figure 2.73 QQ plot showing residual normality

Homoscedasticity: There is no clear pattern among residuals, they appear randomly scattered with equal variability.

```
# Homoscedacity test
plot(density_prediction,3)
```

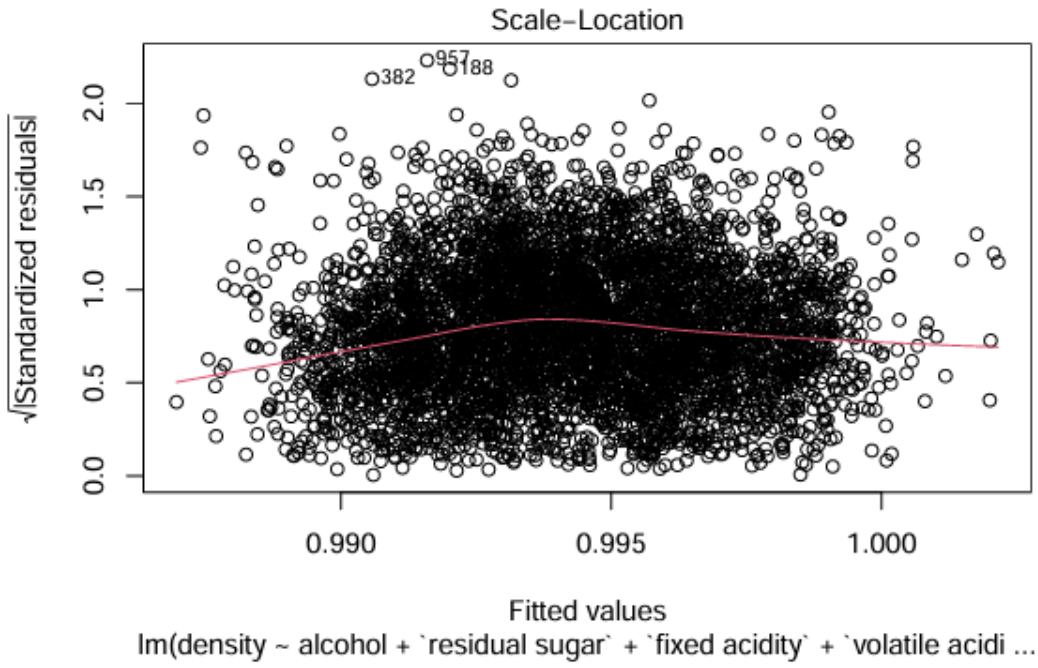


Figure 2.74 Plot showing homoscedasticity

Multicollinearity: The VIF of all the IVs is less than 2, which indicates little to no correlation between the IVs and DV.

1.5.3.2 Result

Based on the regression model, this equation can be used to predict the density of wine.

$$Desnity = 0.998 + 0.0048(volatile\ acidity) + 0.00029(residual\ sugar) + 0.0009(fixed\ acidity) - 0.0012(alcohol)$$

1.6 Findings, Recommendation, and Conclusion

1.6.1 Key Findings

- Alcohol is the strongest predictor of wine quality; higher alcohol content is associated with better quality.
- Volatile acidity negatively impacts quality.
- Higher levels of residual sugar are linked to white wine but show no significant positive effect on quality.
- The hypothesis test shows a significant association between wine type and quality.

1.6.2 Recommendations

1. Optimize wine quality by monitoring and controlling the physicochemical properties to ensure consistency and high quality.
2. Make target adjustments to some physicochemical properties like residual sugar and acidity levels to improve the quality and taste of the wines.
3. As alcohol is the most important predictor of quality, focus on alcohol content during manufacturing to achieve the desired quality.

1.6.3 Conclusion

This analysis provides actionable insights into the drivers of wine quality and the differences between red and white wines. By optimizing properties like alcohol content, residual sugar and volatile acidity, winemakers can improve the overall quality and taste of wines. With these findings, production strategies can be put in place to improve wine quality and increase customer satisfaction.

2 References

Arsham, H. & Lovric, M. (2011). Bartlett's Test. *International Encyclopedia of Statistical Science*. 2. 20-23. 10.1007/978-3-642-04898-2_132.

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., & Reis, J. (2009). Modelling wine preferences by data mining from physicochemical properties. *Decision Support Systems, 47*(4): 547-553. ISSN: 0167-9236.

Kassambara, A. (2018). Classification Method Essential: Logistic Regression Assumptions and Diagnostics. *Statistical tools for high-throughput data analysis.* http://www.sthda.com/english/articles/36-classification-methods-essentials/148-logisticregression-assumptions-and-dignostics-in-r/

Khamis, H. (2008). Measures of Association: How to Choose? *Journal of Diagnostic Medical Sonography*, *24*(3), 155–162. https://doi.org/10.1177/8756479308317006

KORNBROT, D., Howell, D. C., & Everitt, B. S. (2005). Point Biserial Correlation. In *Encyclopedia of Statistics in Behavioral Science* (Vol. 3, pp. 1552–1553). John Wiley & Sons, Ltd. https://doi.org/10.1002/0470013192.bsa485

McAleer, P., (2022). *A Handy Workbook for Research Methods & Statistics* (0.0.9012). Zenodo. https://doi.org/10.5281/zenodo.5934243

McHugh, M. L. (2013). The chi-square test of independence. *Biochemia Medica*, *23*(2), 143–149. https://doi.org/10.11613/BM.2013.018

Nachar, N. (2008). The Mann-Whitney U: A Test for Assessing Whether Two Independent Samples Come from the Same Distribution. *Tutorials in Quantitative Methods for Psychology*, *4*(1), 13–20. https://doi.org/10.20982/tqmp.04.1.p013

Zar, J.H. (1999*) Biostatistical Analysis*. (4th ed.). Prentice Hall, Upper Saddle River.