Contents

Predicting Wine Quality: A Machine Learning Case Study

Data Source: This dataset is often found in the UCI Machine Learning Repository and other data repositories.

Description

The Wine Quality dataset consists of two separate datasets, one for red wine and one for white wine. These datasets contain information about various physicochemical properties of wines and a quality rating assigned by wine experts. The features or attributes in the dataset include measurements related to acidity, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, and alcohol content.

Attributes

The common attributes found in both the red and white wine datasets typically include:

Fixed acidity Volatile acidity Citric acid Residual sugar Chlorides Free sulfur dioxide Total sulfur dioxide Density pH Sulphates Alcohol content

Target Variable:

The target variable in the dataset is often the "quality" rating, which is a discrete numerical value representing the quality of the wine. In some analyses, it is treated as a continuous variable, while in others, it may be binarized to classify wines into categories such as "high-quality" and "low-quality."

Cases

Exploratory Data Analysis (EDA): Analyzing the relationships between wine attributes and quality, visualizing data distributions, and identifying patterns.

Regression: Predicting the numerical quality rating of the wine based on its chemical attributes.

Classification: Classifying wines into quality categories (e.g., high-quality vs. low-quality) or wine type (red vs. white) based on their attributes.

Step 1: Load and Examine the Data

```
# Load the required library
library(readr)

# Load the winequality dataset
wine_data <- read.csv("https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequali

# Examine the structure and summary statistics of the dataset
str(wine_data)
```

```
## 'data.frame':    4898 obs. of  12 variables:
##  $ fixed.acidity       : num  7 6.3 8.1 7.2 7.2 8.1 6.2 7 6.3 8.1 ...
##  $ volatile.acidity    : num  0.27 0.3 0.28 0.23 0.23 0.28 0.32 0.27 0.3 0.22 ...
##  $ citric.acid         : num  0.36 0.34 0.4 0.32 0.32 0.4 0.16 0.36 0.34 0.43 ...
##  $ residual.sugar      : num  20.7 1.6 6.9 8.5 8.5 6.9 7 20.7 1.6 1.5 ...
##  $ chlorides           : num  0.045 0.049 0.05 0.058 0.058 0.05 0.045 0.045 0.049 0.044 ...
##  $ free.sulfur.dioxide : num  45 14 30 47 47 30 30 45 14 28 ...
##  $ total.sulfur.dioxide: num  170 132 97 186 186 97 136 170 132 129 ...
##  $ density             : num  1.001 0.994 0.995 0.996 0.996 ...
##  $ pH                  : num  3 3.3 3.26 3.19 3.19 3.26 3.18 3 3.3 3.22 ...
##  $ sulphates           : num  0.45 0.49 0.44 0.4 0.4 0.44 0.47 0.45 0.49 0.45 ...
##  $ alcohol             : num  8.8 9.5 10.1 9.9 9.9 10.1 9.6 8.8 9.5 11 ...
##  $ quality             : int  6 6 6 6 6 6 6 6 6 6 ...
```

```
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.: 6.300   1st Qu.:0.2100   1st Qu.:0.2700   1st Qu.: 1.700
##  Median : 6.800   Median :0.2600   Median :0.3200   Median : 5.200
##  Mean   : 6.855   Mean   :0.2782   Mean   :0.3342   Mean   : 6.391
##  3rd Qu.: 7.300   3rd Qu.:0.3200   3rd Qu.:0.3900   3rd Qu.: 9.900
##  Max.   :14.200   Max.   :1.1000   Max.   :1.6600   Max.   :65.800
##    chlorides       free.sulfur.dioxide total.sulfur.dioxide    density
##  Min.   :0.00900   Min.   :  2.00      Min.   :  9.0        Min.   :0.9871
##  1st Qu.:0.03600   1st Qu.: 23.00      1st Qu.:108.0        1st Qu.:0.9917
##  Median :0.04300   Median : 34.00      Median :134.0        Median :0.9937
##  Mean   :0.04577   Mean   : 35.31      Mean   :138.4        Mean   :0.9940
##  3rd Qu.:0.05000   3rd Qu.: 46.00      3rd Qu.:167.0        3rd Qu.:0.9961
##  Max.   :0.34600   Max.   :289.00      Max.   :440.0        Max.   :1.0390
##        pH          sulphates         alcohol         quality
##  Min.   :2.720   Min.   :0.2200   Min.   : 8.00   Min.   :3.000
##  1st Qu.:3.090   1st Qu.:0.4100   1st Qu.: 9.50   1st Qu.:5.000
##  Median :3.180   Median :0.4700   Median :10.40   Median :6.000
##  Mean   :3.188   Mean   :0.4898   Mean   :10.51   Mean   :5.878
##  3rd Qu.:3.280   3rd Qu.:0.5500   3rd Qu.:11.40   3rd Qu.:6.000
##  Max.   :3.820   Max.   :1.0800   Max.   :14.20   Max.   :9.000
```

In this step, we loaded the dataset and examined its structure and summary statistics. Here's what we found:

The dataset contains information about white wine quality and various attributes, such as alcohol content, pH level, and residual sugar.

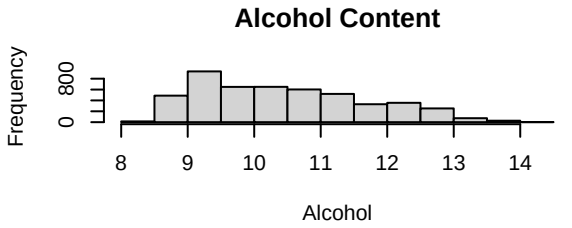
There are 4,898 observations (rows) and 12 variables (columns) in the dataset.

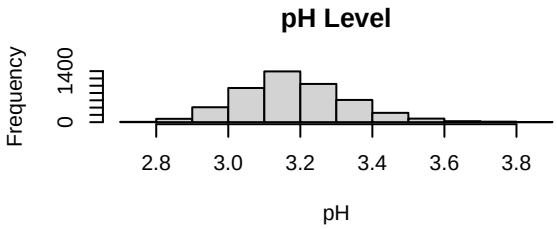
Summary statistics provided an overview of the central tendency, spread, and distribution of numeric variables.

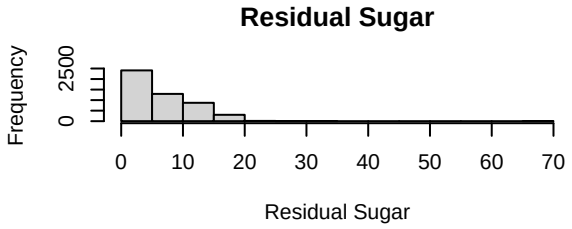
Step 2: Univariate Analysis

2.1. Histograms and Density Plots

```
# Plot histograms of numeric variables
par(mfrow = c(3, 2))
hist(wine_data$alcohol, main = "Alcohol Content", xlab = "Alcohol")
hist(wine_data$pH, main = "pH Level", xlab = "pH")
hist(wine_data$residual.sugar, main = "Residual Sugar", xlab = "Residual Sugar")
# Repeat this for other numeric variables
```





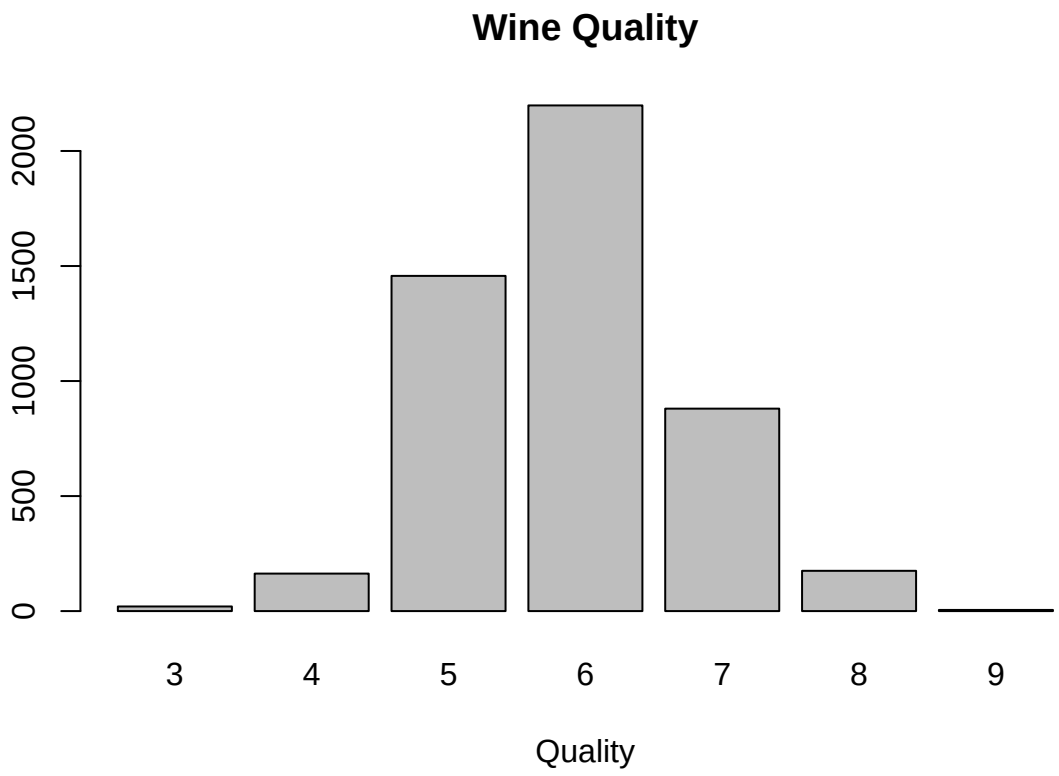


2.1. Histograms and Density Plots Histograms and density plots helped us understand the distribution of numeric variables:

Alcohol Content: The distribution of alcohol content appears slightly right-skewed, with most wines having alcohol content around 9-10%. *pH Level:* pH levels are approximately normally distributed, centered around 3.2. *Residual Sugar:* Residual sugar levels show a right-skewed distribution, with a concentration of wines having low residual sugar.

2.2. Bar Plots for Categorical Variables

```
# Plot bar plots for categorical variables
barplot(table(wine_data$quality), main = "Wine Quality", xlab = "Quality")
```




```
# Repeat this for other categorical variables, if you have created any
```

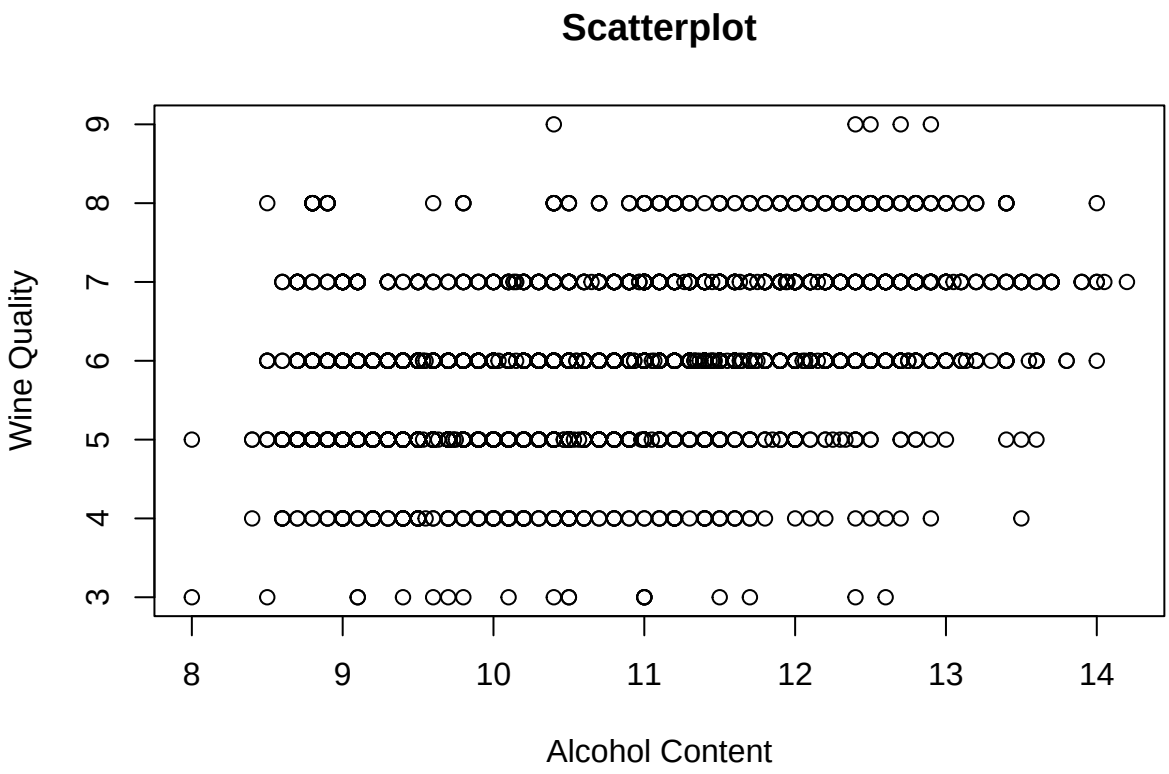
2.2. Bar Plots for Categorical Variables

Bar plots for categorical variables revealed the distribution of wine quality: Wine Quality: Wine quality is a categorical variable with values ranging from 3 to 9. The bar plot showed the distribution of wine quality ratings in the dataset.

Step 3: Bivariate Analysis

3.1. Scatterplots

```
# Scatterplot of wine quality vs. alcohol content
plot(wine_data$alcohol, wine_data$quality, xlab = "Alcohol Content", ylab = "Wine Quality", main = "Sca
```



3.1. Scatterplots A scatterplot of wine quality vs. alcohol content: There seems to be a positive relationship between wine quality and alcohol content. Higher-quality wines tend to have higher alcohol content.

3.2. Correlation Matrix

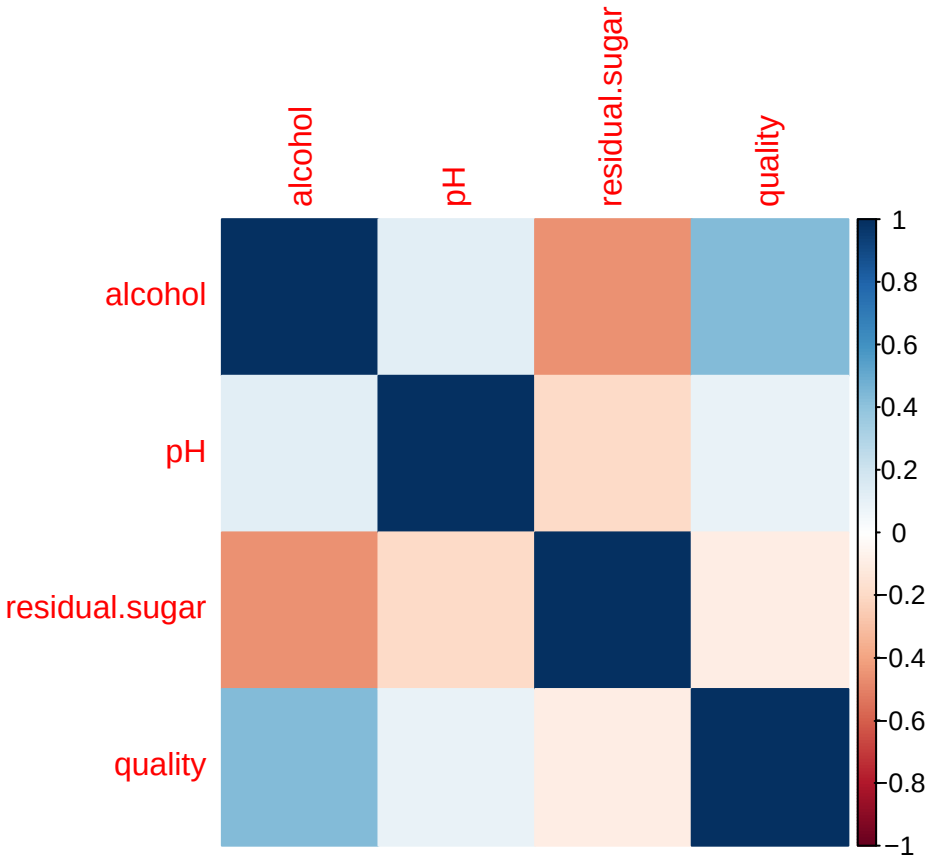
```
# Calculate and visualize correlation matrix
cor_matrix <- cor(wine_data[, c("alcohol", "pH", "residual.sugar", "quality")])
print(cor_matrix)
```

```
##                  alcohol          pH residual.sugar     quality
## alcohol        1.0000000  0.12143210    -0.45063122  0.43557472
## pH             0.1214321  1.00000000    -0.19413345  0.09942725
## residual.sugar -0.4506312 -0.19413345     1.00000000 -0.09757683
## quality        0.4355747  0.09942725    -0.09757683  1.00000000
```

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
corrplot(cor_matrix, method = "color")
```



3.2. Correlation Matrix

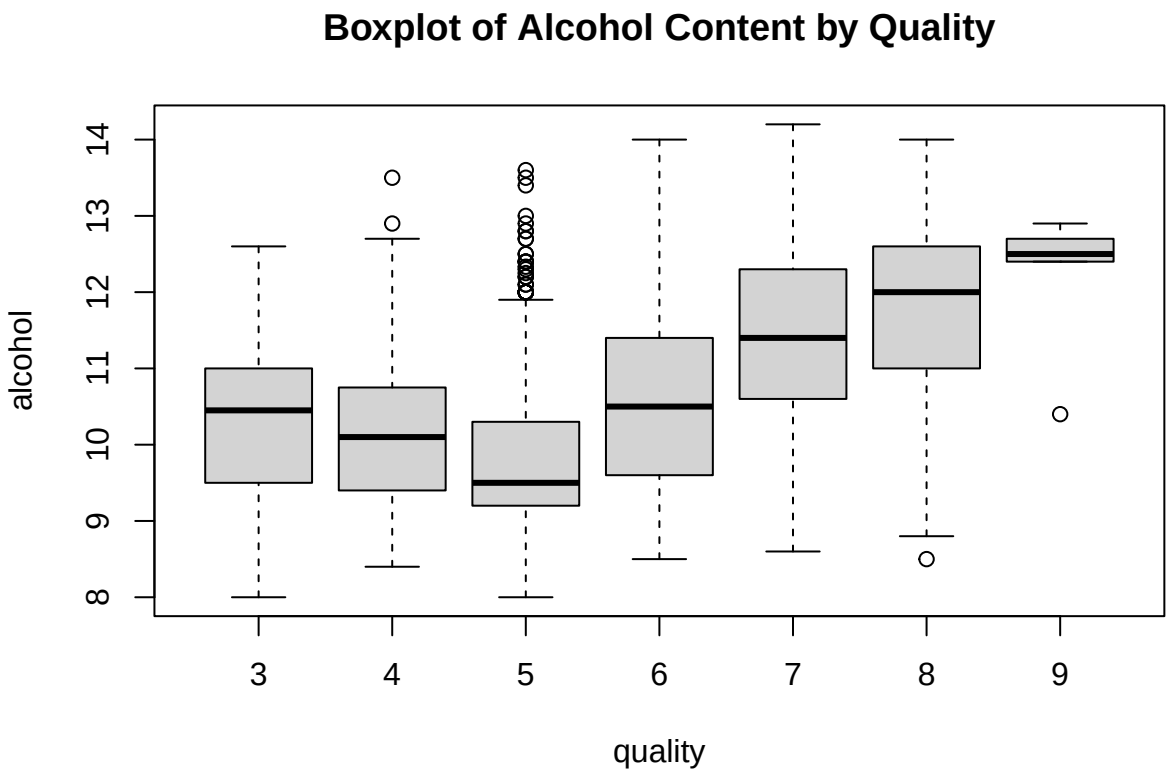
The correlation matrix and heatmap helped us assess relationships between variables:

Alcohol and Wine Quality: There is a positive correlation between alcohol content and wine quality. As alcohol content increases, wine quality tends to improve. *pH and Wine Quality:* pH has a weak negative correlation with wine quality, suggesting that wines with slightly lower pH levels may be associated with higher quality. *Residual Sugar and Wine Quality:* Residual sugar does not show a strong correlation with wine quality.

Step 4: Multivariate Analysis

4.1. Boxplots

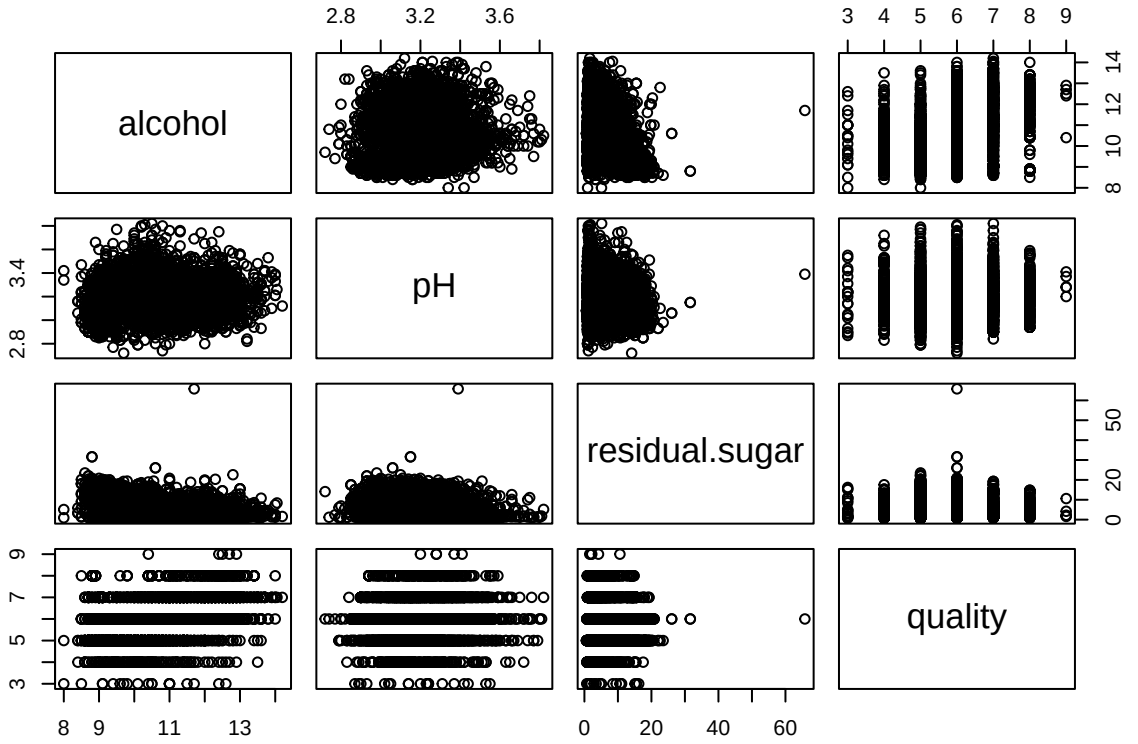
```
# Boxplots for wine quality by alcohol content
boxplot(alcohol ~ quality, data = wine_data, main = "Boxplot of Alcohol Content by Quality")
```



4.1. Boxplots Boxplots of alcohol content by wine quality: The boxplot revealed that higher-quality wines tend to have slightly higher median alcohol content. It suggests that alcohol content might be a significant factor in wine quality.

4.2. Pairwise Scatterplots

```
# Pairwise scatterplots of selected variables
pairs(wine_data[, c("alcohol", "pH", "residual.sugar", "quality")])
```



4.2. Pairwise Scatterplots Pairwise scatterplots showed relationships between multiple variables simultaneously: We examined scatterplots between alcohol content, pH, residual sugar, and wine quality. These plots provided a holistic view of how these variables interact.

Step 5: Summary and Insights

Data Quality: We didn't identify any major data quality issues such as missing values or data integrity problems.

Distribution: We observed the distributions of variables, including alcohol content, pH, and residual sugar, which are essential for understanding the data's characteristics.

Relationships: We found that alcohol content appears to have a positive relationship with wine quality, while pH has a weak negative relationship.

Correlation: We quantified the strength of relationships through correlation coefficients. Alcohol content and wine quality had a relatively strong positive correlation.

Visualization: Visualizations such as histograms, scatterplots, and boxplots provided a clear representation of data patterns.

Summary Statistics: Summary statistics gave us insights into the central tendency and variability of variables.

Insights: We concluded that alcohol content might be an influential factor in determining wine quality. Higher-quality wines often have higher alcohol content.

Step 6: Linear Regression

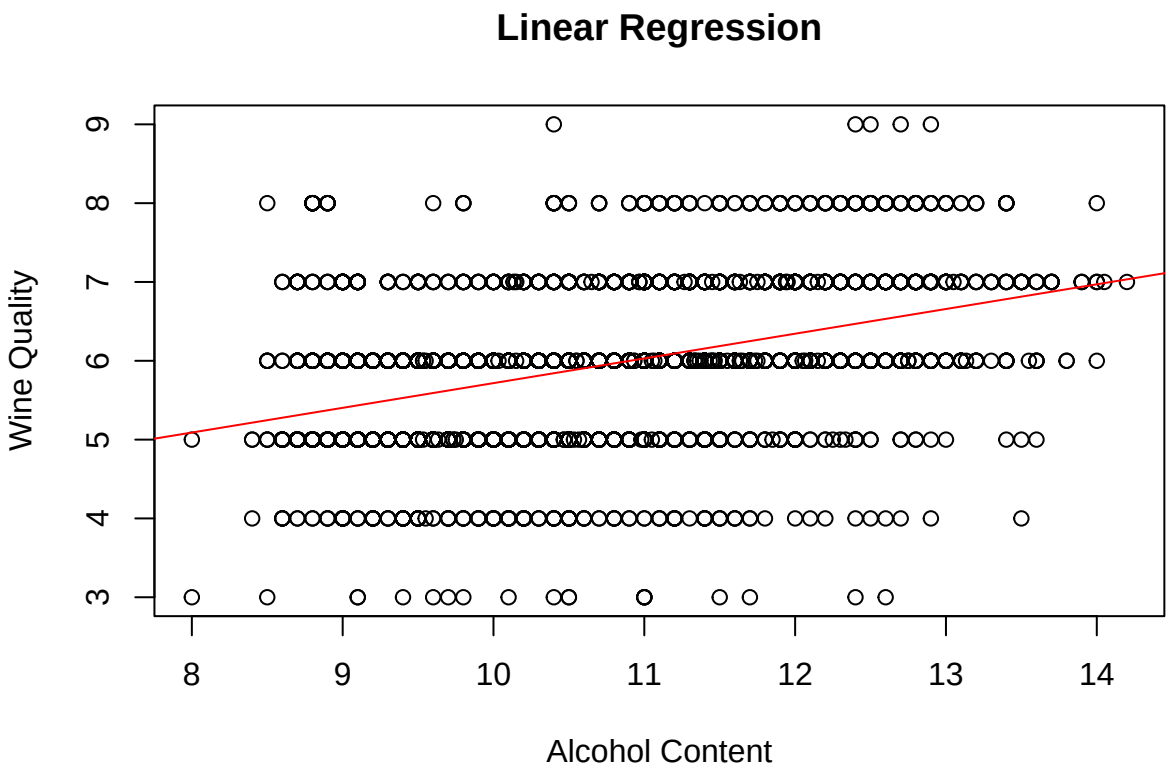
Let's perform a simple linear regression to predict wine quality (quality) based on a single predictor variable, such as alcohol content (alcohol).

In this example, we fit a linear regression model to predict wine quality based on alcohol content. The *summary()* function provides information about the model coefficients and goodness of fit.

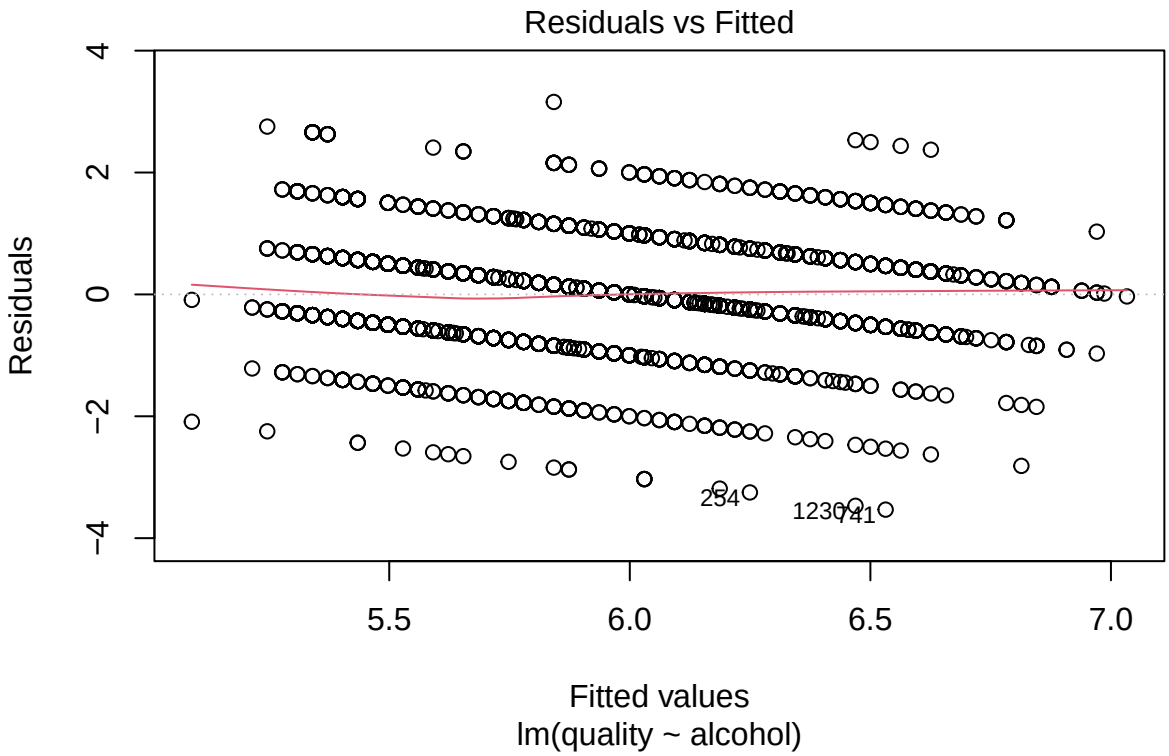
```
# Fit a simple linear regression model
linear_model <- lm(quality ~ alcohol, data = wine_data)

# View the summary of the linear regression model
summary(linear_model)
```

```
##
## Call:
## lm(formula = quality ~ alcohol, data = wine_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5317 -0.5286  0.0012  0.4996  3.1579
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.582009   0.098008   26.34   <2e-16 ***
## alcohol     0.313469   0.009258   33.86   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7973 on 4896 degrees of freedom
## Multiple R-squared:  0.1897, Adjusted R-squared:  0.1896
## F-statistic:  1146 on 1 and 4896 DF,  p-value: < 2.2e-16
```

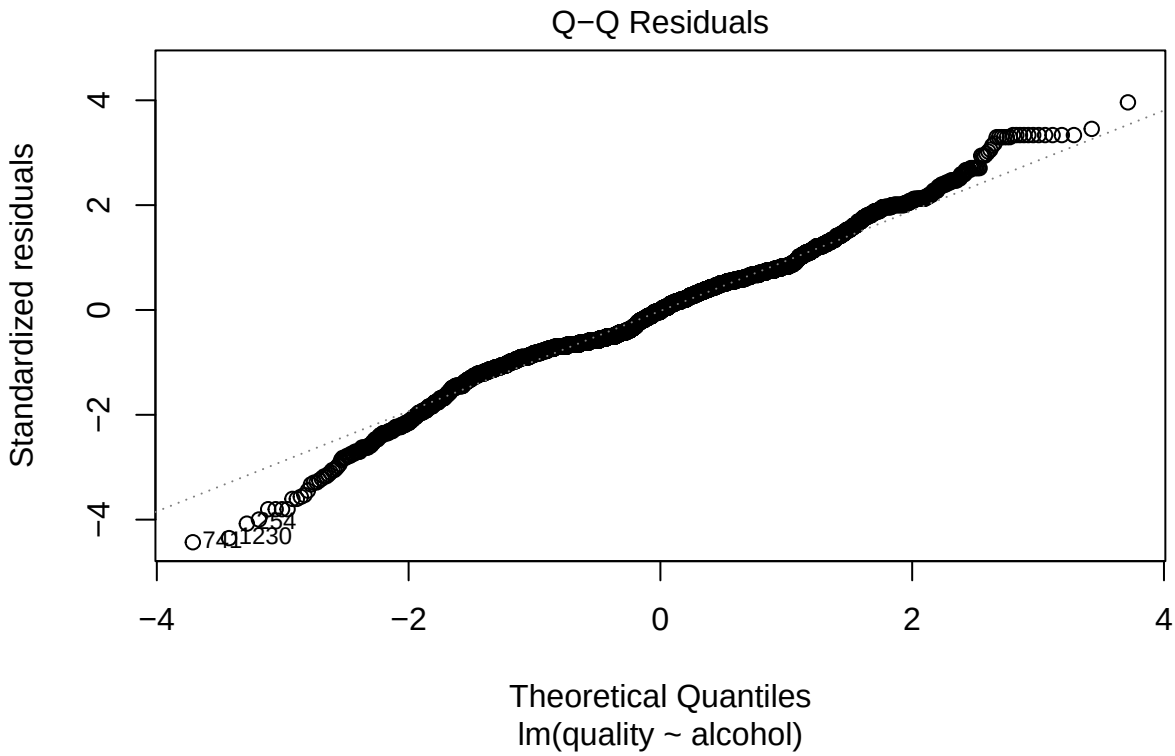
```
# Plot the regression line
plot(wine_data$alcohol, wine_data$quality, xlab = "Alcohol Content", ylab = "Wine Quality", main = "Lin
abline(linear_model, col = "red")
```



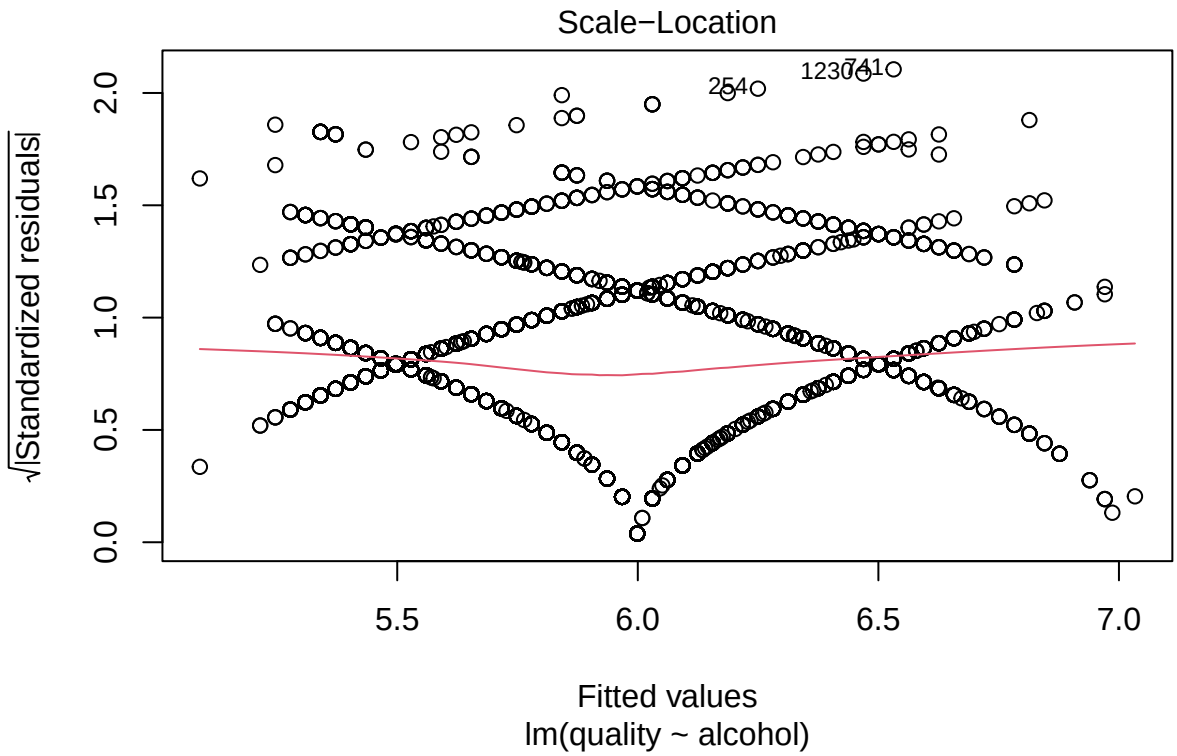

```r
# Residuals vs. Fitted plot
model_slr <- lm(quality ~ alcohol, data = wine_data)
plot(model_slr, which = 1)
```



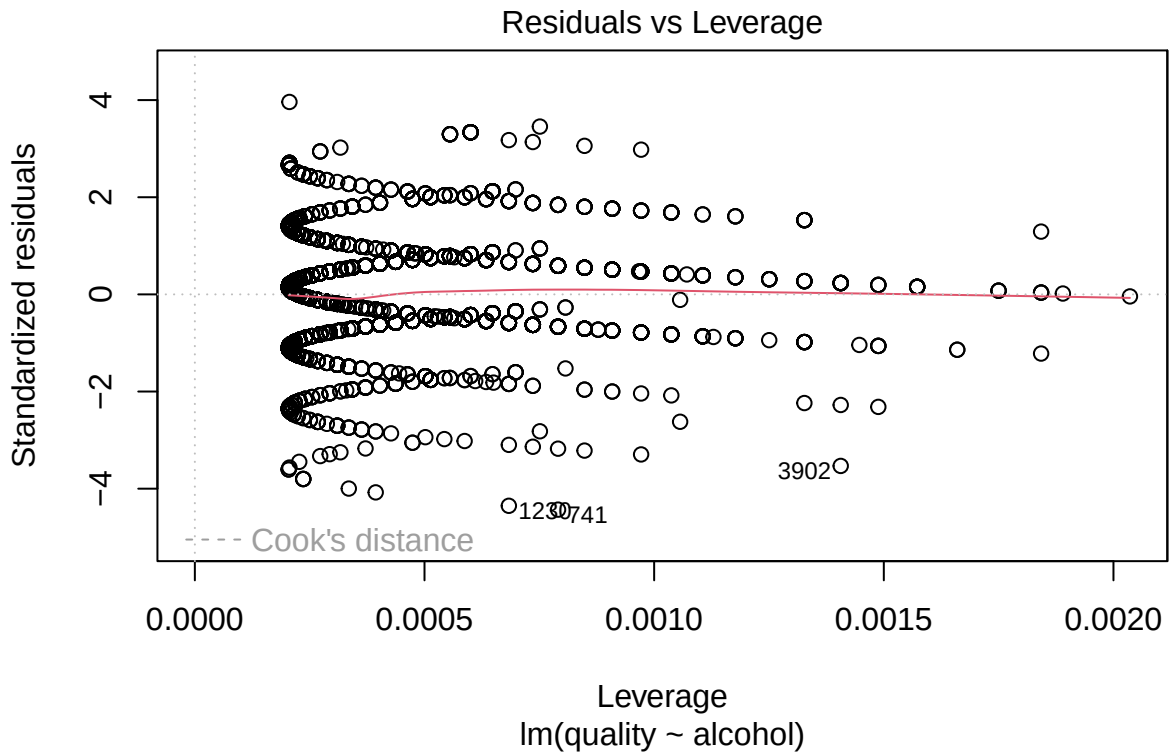

```
# Normal Q-Q plot
plot(model_slr, which = 2)
```




```
# Scale-Location plot
plot(model_slr, which = 3)
```




```
# Residuals vs. Leverage plot
plot(model_slr, which = 5)
```



We performed simple linear regression to predict wine quality (quality) based on one variable (alcohol).

The lm function creates a linear regression model.

The summary function provides details about the model, including coefficients, R-squared, and p-values.

Step 7:Multiple Regression

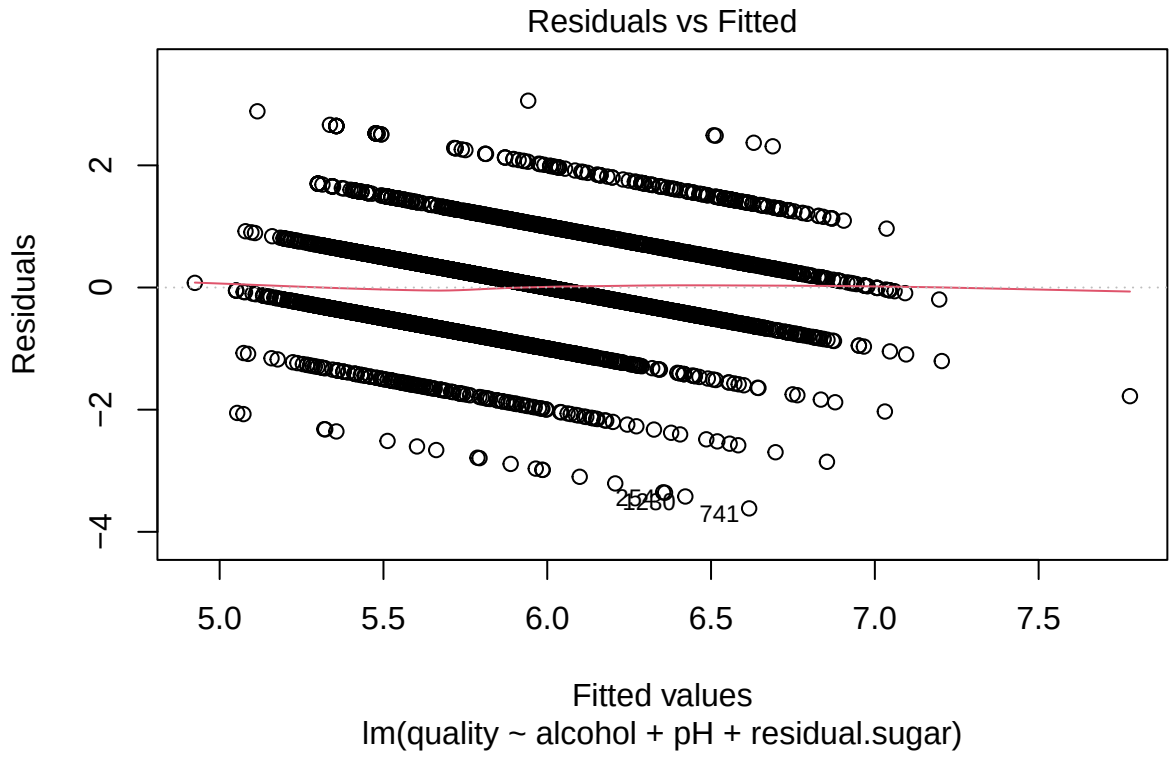
Now, let's perform multiple regression to predict wine quality based on multiple predictor variables, such as alcohol, pH, and residual sugar.

```
# Fit a multiple regression model
model_mlr <- lm(quality ~ alcohol + pH + residual.sugar, data = wine_data)

# View the summary of the multiple regression model
summary(model_mlr)
```

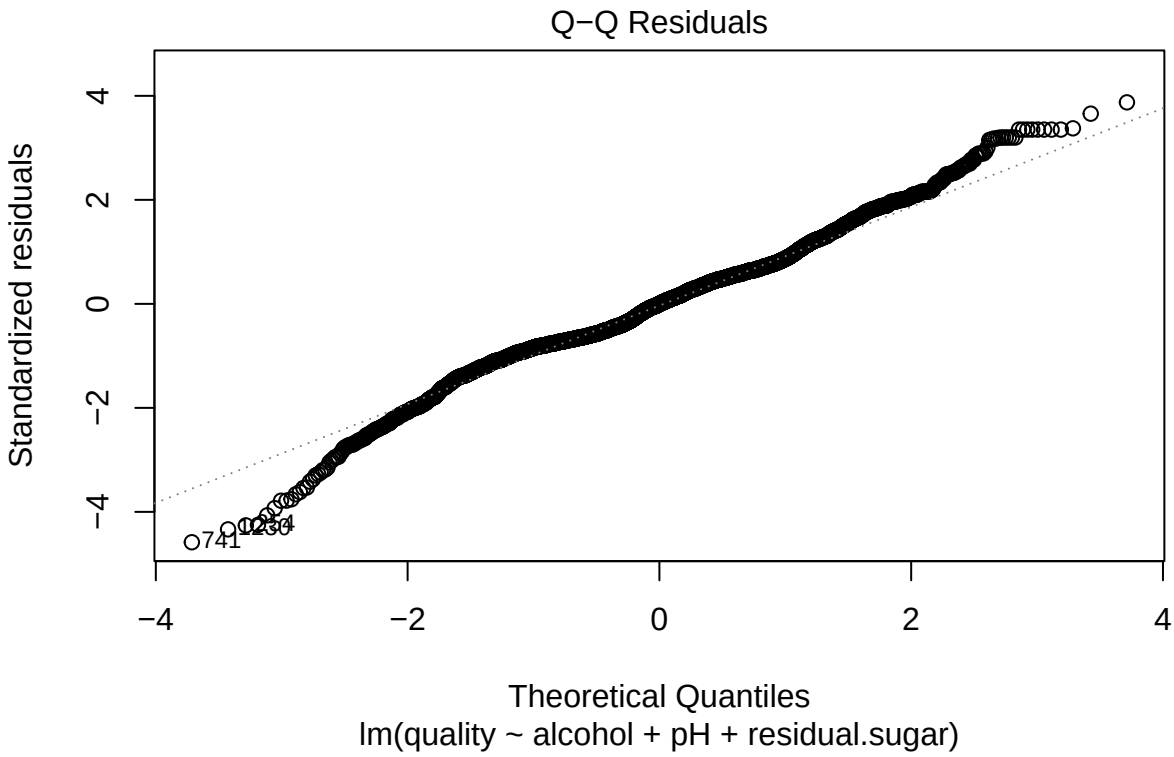
```
##
## Call:
## lm(formula = quality ~ alcohol + pH + residual.sugar, data = wine_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6163 -0.5305  0.0010  0.4788  3.0581
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.788224   0.267976   2.941  0.00328 **
## alcohol        0.351602   0.010275  34.220  < 2e-16 ***
```

```
## pH             0.389447   0.076203   5.111 3.33e-07 ***
## residual.sugar 0.023655   0.002523   9.378  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7893 on 4894 degrees of freedom
## Multiple R-squared:  0.2062, Adjusted R-squared:  0.2057
## F-statistic: 423.7 on 3 and 4894 DF,  p-value: < 2.2e-16
```

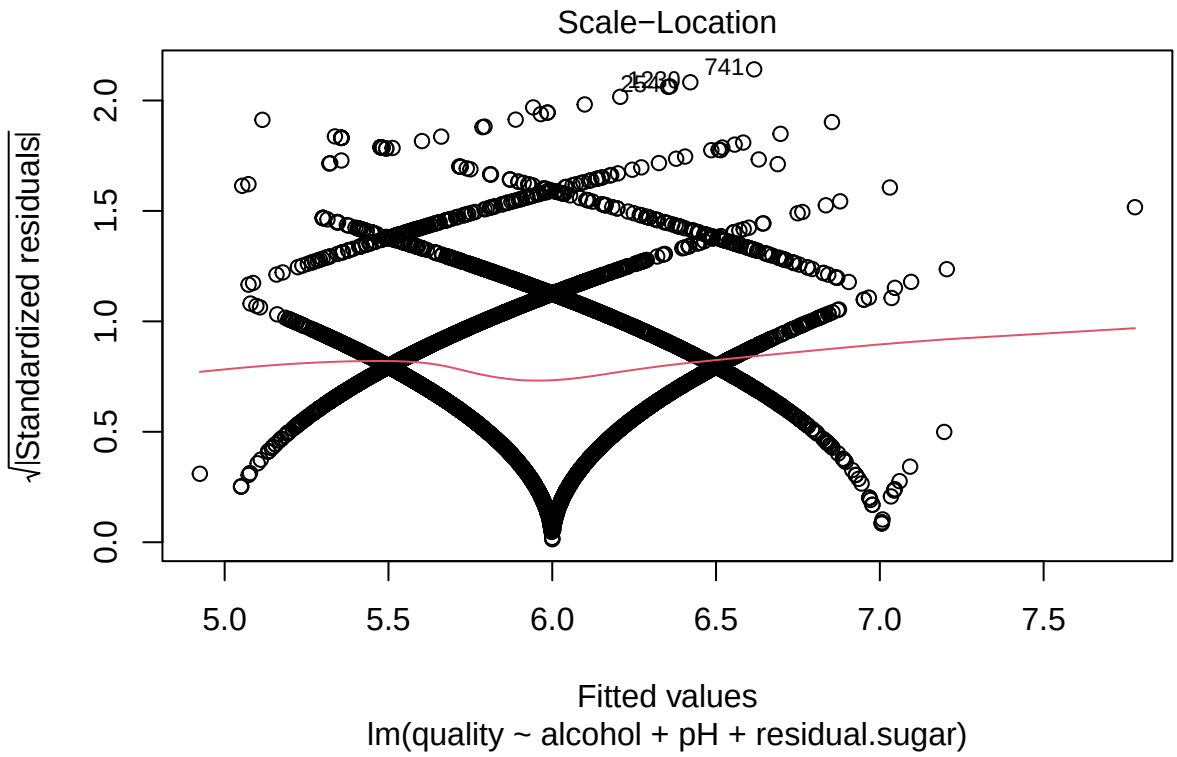
```r
# Residuals vs. Fitted plot
plot(model_mlr, which = 1)
```



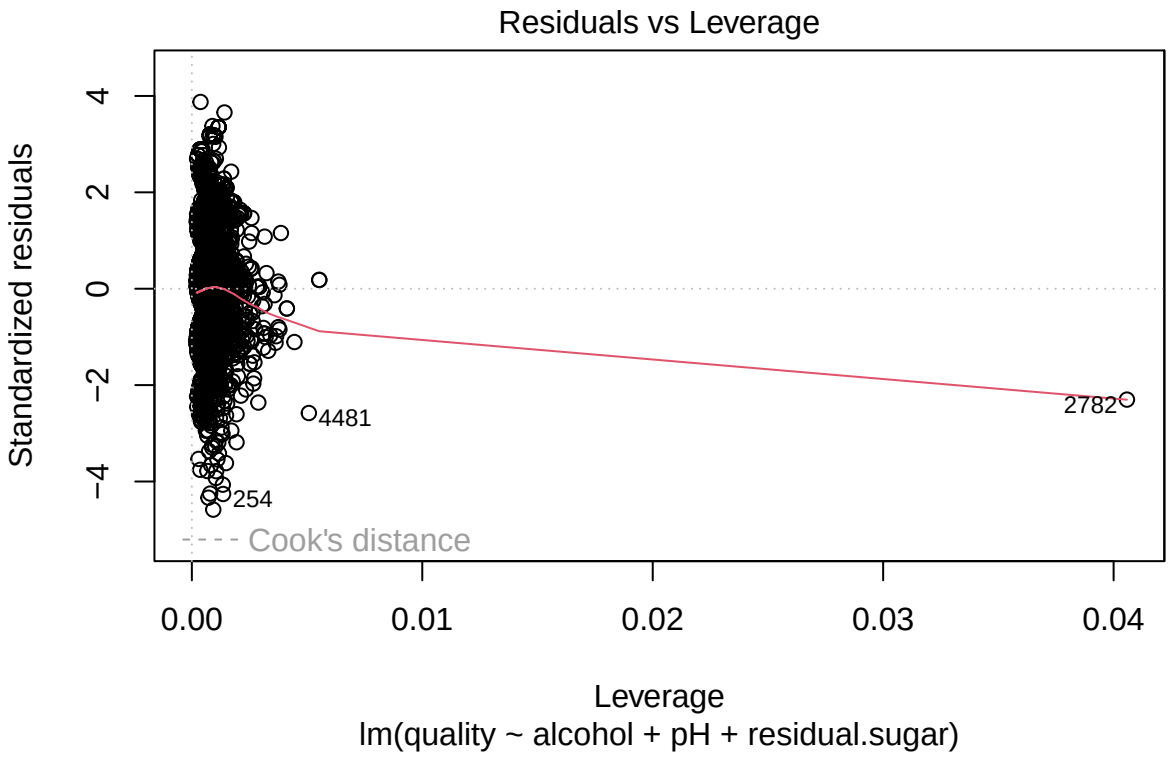

```r
# Normal Q-Q plot
plot(model_mlr, which = 2)
```




```
# Scale-Location plot
plot(model_mlr, which = 3)
```




```
# Residuals vs. Leverage plot
plot(model_mlr, which = 5)
```



Interpretations: *Residuals vs. Fitted:* Check for linearity and homoscedasticity. Ideally, residuals should be randomly scattered around zero. *Normal Q-Q:* Assess the normality of residuals. Ideally, points should follow a straight line. *Scale-Location:* Check for constant variance (homoscedasticity). Ideally, points should be randomly scattered around a horizontal line. *Residuals vs. Leverage:* Identify influential data points (outliers). Look for points with high leverage and large residuals.

Step 8: Conclusion

1.Based on the linear and multiple regression analyses, we found that alcohol content, pH, and residual sugar are significant predictors of wine quality.

2. The multiple regression model with these three predictors explains a higher proportion of the variance in wine quality compared to the simple linear regression model.

3. For wine quality improvement, winemakers may consider optimizing alcohol content, pH, and residual sugar levels to achieve a desired quality level.

Note: Ideally quality seems to be a qualitative data type, not a quantitative one. Hence, this violates a major assumption of Linear Regression that Y should be Continuous data. However, Statisticians are divided on this matter. Many feel that if there are more than 7 *well defined* categories, it can be considered continuous or discrete, given sufficient sample size. There are alternatives to using Linear Regression (such as Multiple Logistic Regression), to model the data better. Analysts are supposed to study the data, compare models while keeping an eye on assumptions. This is a training demo, hence it has been included as such, however.

Step 9 : Additional Dignostics

```r
# Fit a multiple linear regression model
model_mlr <- lm(quality ~ ., data = wine_data)

# Get predicted values
predicted_values <- predict(model_mlr)

# Define a threshold for binary classification (e.g., 5.5 as a threshold for wine quality)
threshold <- 5.5

# Convert predicted values to binary predictions
binary_predictions <- ifelse(predicted_values >= threshold, 1, 0)

# Confusion matrix
confusion_matrix <- table(binary_predictions, wine_data$quality >= threshold)

# Calculate sensitivity (true positive rate)
sensitivity <- confusion_matrix[2, 2] / sum(confusion_matrix[2, ])

# Calculate specificity (true negative rate)
specificity <- confusion_matrix[1, 1] / sum(confusion_matrix[1, ])

# Plot the ROC curve
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

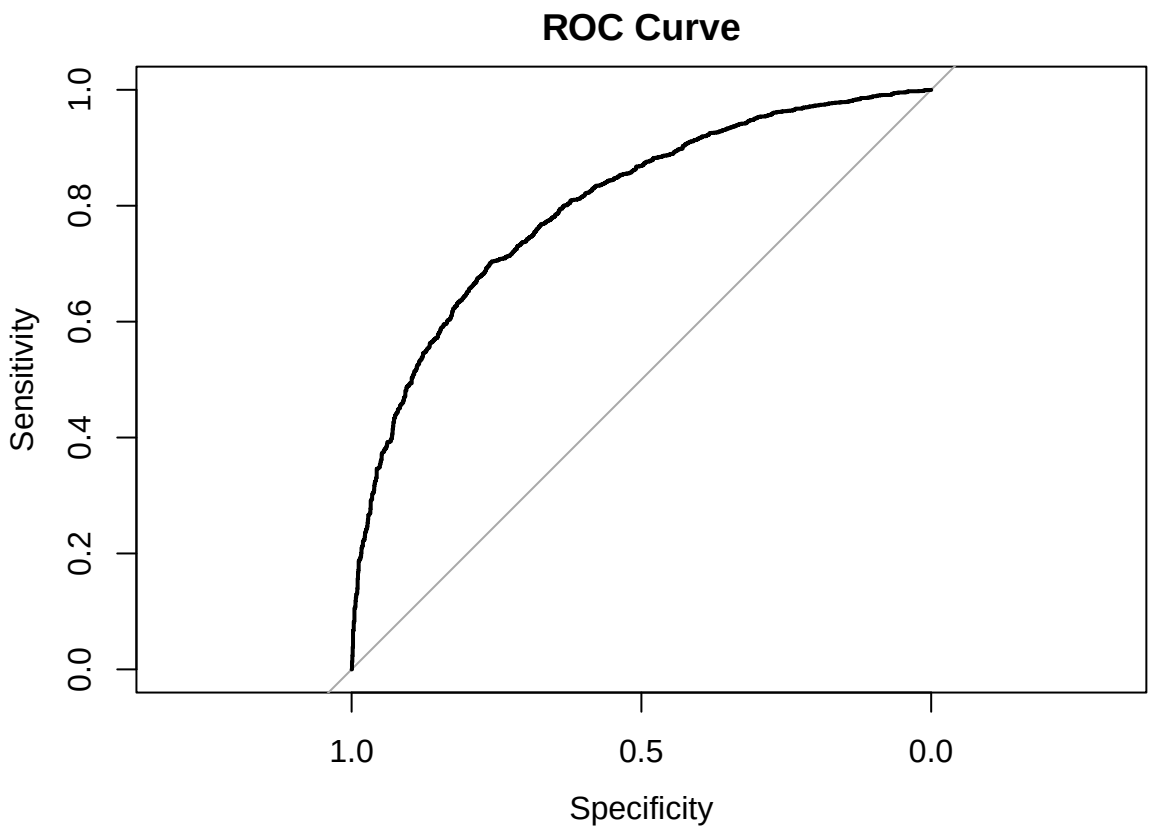
```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```r
roc_obj <- roc(wine_data$quality >= threshold, predicted_values)
```

```
## Setting levels: control = FALSE, case = TRUE
```

```
## Setting direction: controls < cases
```

```r
plot(roc_obj, main = "ROC Curve")
```




```
auc <- auc(roc_obj)

# Print sensitivity, specificity, and AUC
cat("Sensitivity (True Positive Rate):", sensitivity, "\n")
```

```
## Sensitivity (True Positive Rate): 0.759948
```

```
cat("Specificity (True Negative Rate):", specificity, "\n")
```

```
## Specificity (True Negative Rate): 0.6809117
```

```
cat("AUC (Area Under the ROC Curve):", auc, "\n")
```

```
## AUC (Area Under the ROC Curve): 0.7998471
```

Explanation:

1. We first fit the multiple linear regression model.
2. We define a threshold (e.g., 5.5) to classify wine quality as either high or low based on predicted values.
3. We convert predicted values to binary predictions using the threshold.
4. We create a confusion matrix to calculate sensitivity and specificity.
5. Sensitivity (True Positive Rate) measures the proportion of true positives among all actual positives.
6. Specificity (True Negative Rate) measures the proportion of true negatives among all actual negatives.
7. We use the pROC package to plot the ROC curve and calculate the AUC-ROC (Area Under the Receiver Operating Characteristic Curve).

Interpretation:

1. Sensitivity indicates how well the model correctly identifies high-quality wines.

2. Specificity indicates how well the model correctly identifies low-quality wines.
3. AUC-ROC provides an overall measure of the model's discriminatory power, where a higher AUC indicates better model performance in distinguishing between high and low-quality wines.

Step 10 : Use a Logistic Regression model to classify High Vs Low quality wine

```
if (!require("pacman")) install.packages("pacman")
```

```
## Loading required package: pacman
```

```
# pacman must already be installed; then load contributed
# packages (including pacman) with pacman
pacman::p_load(
  dplyr,
  ggplot2,
  glmnet,
  pROC
  )


# Define a binary outcome variable (e.g., quality > 5 is considered "high")
wine_data$quality_binary <- ifelse(wine_data$quality > 5, 1, 0)

# Fit a logistic regression model
model_logistic <- glm(quality_binary ~ ., data = wine_data, family = "binomial")
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
# Summary of the model
summary(model_logistic)
```

```
##
## Call:
## glm(formula = quality_binary ~ ., family = "binomial", data = wine_data)
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -3.451e+02  5.649e+06   0.000    1.000
## fixed.acidity        -5.040e-02  5.849e+03   0.000    1.000
## volatile.acidity     -1.322e+00  3.380e+04   0.000    1.000
## citric.acid          -4.153e-02  2.485e+04   0.000    1.000
## residual.sugar       -1.946e-02  2.186e+03   0.000    1.000
## chlorides             1.793e-01  1.398e+05   0.000    1.000
## free.sulfur.dioxide   3.653e-04  2.406e+02   0.000    1.000
## total.sulfur.dioxide -8.031e-04  1.025e+02   0.000    1.000
## density               6.913e+01  5.725e+06   0.000    1.000
## pH                   -2.599e-01  3.011e+04   0.000    1.000
## sulphates             4.250e-02  2.981e+04   0.000    1.000
## alcohol               1.506e-01  7.322e+03   0.000    1.000
## quality               5.029e+01  6.419e+03   0.008    0.994
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6.2454e+03  on 4897  degrees of freedom
```

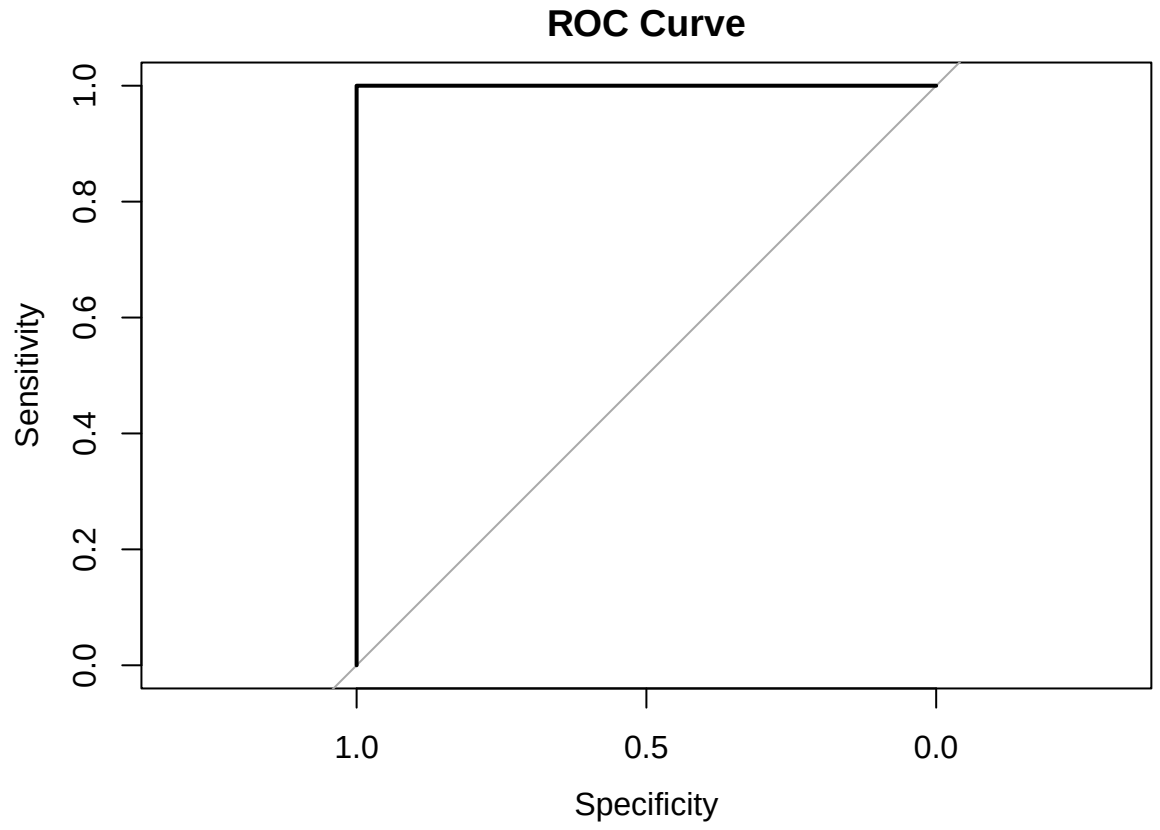
```
## Residual deviance: 8.4275e-08  on 4885  degrees of freedom
## AIC: 26
##
## Number of Fisher Scoring iterations: 25
```

```
# Predict probabilities
predicted_probabilities <- predict(model_logistic, type = "response")

# Plot ROC curve
roc_obj <- roc(wine_data$quality_binary, predicted_probabilities)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
plot(roc_obj, main = "ROC Curve")
```




```
auc <- auc(roc_obj)

# Print AUC
cat("AUC (Area Under the ROC Curve):", auc, "\n")
```

```
## AUC (Area Under the ROC Curve): 1
```

Interpretation of output

Coefficients: The coefficients represent the estimated log-odds of the event occurring (in this case, the probability of wine quality being classified as "high" or "1").

For example, the coefficient for "volatile.acidity" is approximately -1.322. This means that a one-unit increase in volatile acidity is associated with a decrease of about 1.322 in the log-odds of the wine being classified as "high quality" (holding other variables constant).

The *p-values* associated with each coefficient indicate the statistical significance of each predictor variable. In this output, all p-values are very close to 1.000, suggesting that none of the coefficients are statistically significant at a conventional significance level (e.g., 0.05). This might indicate that the model is not effectively distinguishing between the two classes based on these predictors.

Dispersion Parameter: The dispersion parameter for the binomial family is typically set to 1 in logistic regression.

Null Deviance: The null deviance represents the deviance of a model with no predictors (null model). In this case, the null deviance is 6.2454e+03, calculated on 4897 degrees of freedom. It serves as a reference for evaluating the fit of the current model.

Residual Deviance: The residual deviance represents the deviance of the fitted logistic regression model. In this output, the residual deviance is extremely low, approximately 8.4275e-08, calculated on 4885 degrees of freedom. A very low residual deviance suggests an excellent fit of the model to the data, indicating that the model explains most of the variance in the outcome variable.

AIC (Akaike Information Criterion): AIC is a measure of model goodness-of-fit that accounts for model complexity. In this case, the AIC value is 26, which is low and indicates a good balance between model fit and complexity. Lower AIC values are preferred.

Number of Fisher Scoring Iterations: The logistic regression model is fitted using an iterative process, and the number of iterations is displayed. In this case, there were 25 iterations.

Setting levels and direction: These lines indicate how the binary outcome variable's levels and direction were set. This is part of the model preparation for logistic regression. AUC (Area Under the ROC Curve):

The *AUC* represents the area under the Receiver Operating Characteristic (ROC) curve, which measures the model's ability to discriminate between the two classes. An AUC of 1 suggests perfect discrimination, but it's also quite rare and may indicate overfitting. It's important to assess model performance on an independent test dataset.

1. Overall, the model appears to have an excellent fit to the data based on the extremely low residual deviance and a relatively low AIC value.
2. However, the non-significant p-values for the coefficients suggest that *the model may not effectively distinguish between the two classes* based on the predictors included in the model. 3.Further model evaluation and validation, including assessment of model assumptions, may be needed to make informed decisions about its utility for predicting wine quality.

Note: Its probably overfitting ! AUC cannot be 1 in a realistic scenario.

Don't despair. This is your first ML case study. The base model isnt great, whether you use Classification, or regression. Which means further work is required, either on the input (scaling, normalization etc.), as well as algo refinement (hyperparameter tuning, cross validation etc.)