University of Applied Sciences

HOCHSCHULE EMDEN·LEER

Introduction to Data Sciences

Wine Dataset Analysis

Author:	Hemanth Jadiswami Prabhakaran
Matriculation No.:	7026000
Author:	Manoj Kumar Prabhakaran
Matriculation No.:	7026006
Course of Studies:	MBIDA

First examiner:	Prof. Dr. Joachim Schwarz
Submission date:	June 12, 2025

University of Applied Sciences Emden/Leer · Faculty of Technology · Mechanical Engineering Department
Constantiaplatz 4 · 26723 Emden · http://www.hs-emden-leer.de

Contents

List of Figures

List of Tables

Acronyms

1 Introduction

1.1 Motivation

The wine industry represents a significant economic sector globally, with quality assessment serving as a critical factor in market positioning and consumer satisfaction. Traditional wine quality evaluation relies heavily on expert sensory analysis, which, while valuable, can be subjective and inconsistent. The advent of analytical chemistry and data science methodologies offers new opportunities to develop objective, reproducible approaches to wine quality assessment.

The Portuguese wine industry, with its rich viticultural heritage and diverse terroir, produces wines of varying characteristics and quality levels. Understanding the relationship between chemical composition and perceived quality can provide valuable insights for winemakers, quality control specialists, and researchers in enology.

This study leverages advanced statistical and machine learning techniques to analyze a comprehensive dataset of Portuguese wines, examining both red and white varieties. By applying methods taught in the Introduction to Data Science course, including exploratory data analysis, hypothesis testing, regression analysis, classification techniques, and dimensionality reduction, we aim to uncover patterns and relationships that can inform wine production and quality assessment practices.

1.2 Problem Definition

The primary research problem addresses the predictability of wine quality based on physicochemical properties. Specifically, this study investigates:

1. **Descriptive Analysis**: What are the characteristic chemical profiles of Portuguese red and white wines, and how do these distributions inform our understanding of wine composition?

2. **Comparative Analysis**: Do red and white wines exhibit significantly different alcohol content levels, and what implications does this have for wine classification?

3. **Predictive Modeling**: To what extent can wine quality be predicted from chemical and sensory variables, and which factors are most influential in determining quality ratings?

4. **Classification Performance**: How effectively can wines be classified into quality categories (good vs. bad) and variety types (red vs. white) using chemical composition data?

5. **Dimensionality Reduction**: Can the complex chemical profile of wines be reduced to a smaller set of underlying factors that capture the essential characteristics of wine composition?

These research questions are addressed through the systematic application of statistical methods, ensuring both theoretical rigor and practical relevance to the wine industry.

1.3 Organization of the Rest of the Paper

The remainder of this paper is structured as follows:

Chapter 2 provides the theoretical foundations necessary for understanding wine quality assessment, statistical methods in food science, and machine learning applications in agricultural contexts.

Chapter 3 reviews the current state of research in wine quality prediction, chemical analysis in viticulture, and classification techniques used in the food industry, establishing the academic context for this study.

Chapter 4 formally presents the research hypotheses and statistical formulations that guide the empirical analysis.

Chapter 5 constitutes the core empirical contribution, presenting detailed results from six analytical tasks: exploratory data analysis, hypothesis testing, linear regression, classification methods, predictive modeling with validation, and factor analysis.

Chapter 6 concludes with a comprehensive summary of findings and discusses implications for future research and practical applications in the wine industry.

2 Theoretical Foundations

2.1 Wine Quality Assessment

Wine quality assessment represents a complex intersection of sensory evaluation, chemical analysis, and consumer preference research. Traditionally, wine quality has been evaluated through expert panel tastings, which assess attributes such as appearance, aroma, taste, and overall impression [**jackson2020**]. However, these methods, while comprehensive, suffer from inherent subjectivity and variability between assessors.

The development of analytical chemistry techniques has enabled objective measurement of wine composition, including alcohol content, acidity levels, residual sugars, and various chemical compounds that influence sensory characteristics. The relationship between chemical composition and sensory perception forms the foundation for predictive quality models [**waterhouse2016**].

Quality scores in wine assessment typically follow ordinal scales, with ratings from 0-10 or 0-100 points being common. These scores attempt to quantify overall wine quality but represent subjective evaluations that may vary across different tasting panels and cultural contexts.

2.2 Statistical Methods in Food Science

Statistical analysis in food science encompasses descriptive statistics for characterizing food properties, inferential statistics for hypothesis testing, and multivariate techniques for pattern recognition and classification [**granato2018**].

Descriptive Statistics provide fundamental insights into food composition, including measures of central tendency, variability, and distribution shape. Skewness analysis is particularly relevant for chemical composition data, which often exhibits non-normal distributions.

Hypothesis Testing enables researchers to make inferences about population parameters based on sample data. In wine research, t-tests are commonly used to compare characteristics between different wine types or production methods.

Regression Analysis allows for the modeling of relationships between independent variables (chemical properties) and dependent variables (quality ratings). Multiple regression techniques can identify the most influential chemical factors affecting quality.

Classification Methods include logistic regression, discriminant analysis, and machine learning algorithms that can categorize wines based on chemical profiles. These methods are essential for developing automated quality assessment systems.

2.3 Machine Learning Applications in Agriculture

The application of machine learning techniques in agricultural and food science contexts has grown significantly in recent years [**liakos2018**]. These methods offer powerful tools for pattern recognition, prediction, and classification tasks that traditional statistical approaches may not handle effectively.

Supervised Learning techniques, such as logistic regression and support vector machines, use labeled training data to build predictive models. In wine research, these methods can predict quality categories or wine types based on chemical composition.

Dimensionality Reduction techniques, including Principal Component Analysis (PCA) and Factor Analysis, help identify underlying patterns in high-dimensional chemical data. These methods are particularly valuable for understanding the complex relationships between multiple chemical variables.

Model Validation procedures, including train-test splits and cross-validation, ensure that predictive models generalize well to new data. ROC analysis and AUC metrics provide standardized measures of classification performance.

3 State of Research

3.1 Wine Quality Prediction Studies

Recent literature demonstrates significant interest in developing predictive models for wine quality assessment. Cortez et al. [**cortez2009**] pioneered the use of machine learning techniques on Portuguese wine data, achieving moderate success in predicting quality ratings from physicochemical properties. Their work established the foundation for subsequent research in this domain.

Gupta [**gupta2018**] applied various machine learning algorithms to wine quality prediction, comparing the performance of random forests, support vector machines, and neural networks. The study found that ensemble methods generally outperformed individual algorithms, with chemical acidity and alcohol content being among the most important predictive factors.

More recent research by Kumar et al. [**kumar2020**] explored deep learning approaches for wine quality assessment, achieving improved prediction accuracy compared to traditional methods. However, the authors noted that the interpretability of deep learning models remains a challenge for practical applications in the wine industry.

3.2 Chemical Analysis in Viticulture

The relationship between wine chemistry and quality has been extensively studied in enological research. Ribéreau-Gayon et al. [**ribereau2017**] provide a comprehensive overview of wine chemistry, highlighting the importance of compounds such as phenolics, organic acids, and volatile compounds in determining wine quality and character.

Specific chemical parameters have been identified as quality indicators. Volatile acidity, primarily acetic acid, is generally associated with wine defects when present at elevated levels [**jackson2020**]. Conversely, appropriate levels of fixed acidity contribute to wine structure and stability.

Sulfur dioxide management represents a critical aspect of wine production, with both free and total sulfur dioxide levels requiring careful monitoring to prevent oxidation while avoiding excessive sulfur character [**waterhouse2016**].

3.3 Classification Techniques in Food Industry

Classification methods have found widespread application in food quality assessment and authenticity testing. Downey et al. [**downey2006**] demonstrated the use of spec-

troscopic techniques combined with chemometric analysis for wine origin classification, achieving high accuracy in distinguishing wines from different geographical regions.

Logistic regression has proven particularly effective for binary classification tasks in food science, such as distinguishing between acceptable and unacceptable products based on quality parameters [**granato2018**]. The method's interpretability makes it valuable for regulatory applications where decision reasoning must be transparent.

Factor analysis and principal component analysis have been widely used to understand the underlying structure of complex food composition data [**jolliffe2016**]. These techniques help identify the most important chemical factors that contribute to food quality and characteristics.

4 Research Hypotheses

4.1 Primary Research Questions

Based on the theoretical foundations and literature review, this study addresses the following primary research questions:

RQ1: What are the characteristic distributions and relationships among chemical and sensory variables in Portuguese red and white wines?

RQ2: Do red and white wines exhibit significantly different alcohol content levels?

RQ3: Can wine quality be effectively predicted from chemical and sensory properties using linear regression techniques?

RQ4: How accurately can wines be classified into quality categories (good vs. bad) based on their chemical composition?

RQ5: Can wine variety (red vs. white) be predicted from chemical properties alone, and what is the predictive performance on validation data?

RQ6: What underlying factor structure exists in the chemical composition data, and how many factors are needed to adequately represent wine chemistry?

4.2 Statistical Hypotheses

Hypothesis 1 (H1): Alcohol Content Comparison

- H_0: $\mu_{red} = \mu_{white}$ (No difference in mean alcohol content between red and white wines)
- H_1: $\mu_{red} \neq \mu_{white}$ (Significant difference in mean alcohol content between wine types)
- $\alpha = 0.05$

Hypothesis 2 (H2): Linear Regression Model Significance

- H_0: $\beta_1 = \beta_2 = \ldots = \beta_k = 0$ (No linear relationship between chemical variables and wine quality)
- H_1: At least one $\beta_i \neq 0$ (Significant linear relationship exists)
- $\alpha = 0.05$

Hypothesis 3 (H3): Classification Performance

- H_0: Classification accuracy ≤ 0.5 (No better than random chance)
- H_1: Classification accuracy > 0.5 (Better than random classification)

Hypothesis 4 (H4): Factor Analysis Suitability

- H_0: Correlation matrix is not suitable for factor analysis (KMO < 0.5)
- H_1: Correlation matrix is suitable for factor analysis (KMO ≥ 0.5)

5 Own Empirical Study

5.1 Dataset Description

The analysis utilizes a dataset of Portuguese wines containing 6,497 observations across 13 variables. The dataset includes both red and white wine varieties, with each observation representing a unique wine sample analyzed for chemical composition and rated for quality by expert panels.

Listing 5.1: Data Import and Initial Setup

```
# Read the wine dataset
wine_data <- read.csv("wine copy.csv",
    stringsAsFactors = FALSE)

# Remove the index column (X) as specified in
    assignment
wine_data <- wine_data[, -1]

# Display basic information about the dataset
cat("=== WINE DATASET OVERVIEW ===\n")
cat("Dataset dimensions:", dim(wine_data), "\n")
cat("Number of observations:", nrow(wine_data), "\n")
cat("Number of variables:", ncol(wine_data), "\n\n")

# Convert variety to factor
wine_data$variety <- as.factor(wine_data$variety)
```

5.1.1 Variable Descriptions

Chemical Variables:

- **Fixed Acidity (g/L):** Non-volatile acids that do not evaporate readily
- **Volatile Acidity (g/L):** Acetic acid content, associated with vinegar taste at high levels
- **Citric Acid (g/L):** Adds freshness and flavor in small quantities
- **Residual Sugar (g/L):** Remaining sugar after fermentation completion

- **Chlorides (g/L):** Salt content in wine
- **Free Sulfur Dioxide (mg/L):** Prevents microbial growth and oxidation
- **Total Sulfur Dioxide (mg/L):** Combined free and bound SO_2 forms
- **Density (g/mL):** Wine density relative to water
- **pH:** Acidity/basicity measure on 0-14 scale
- **Sulphates (g/L):** Wine additive contributing to SO_2 levels
- **Alcohol (% vol):** Ethanol content by volume

Target Variables:

- **Quality:** Expert rating on 0-10 scale (higher = better quality)
- **Variety:** Wine type (red or white)

5.2 Exploratory Data Analysis

5.2.1 Summary Statistics

Listing 5.2: Summary Statistics for Metric Variables

```
# Select numeric variables (exclude variety)
numeric_vars <- wine_data[, sapply(wine_data, is.numeric)]

# Comprehensive summary statistics
summary_stats <- describe(numeric_vars)
print(summary_stats)
```

[Insert R output from describe() function showing mean, standard deviation, median, quartiles, min, max for all numeric variables]

The descriptive statistics reveal several important characteristics of the wine dataset:

Central Tendency: Most chemical variables show reasonable distributions around their means, with quality ratings averaging approximately [X.X] on the 10-point scale.

Variability: Standard deviations indicate moderate variability in most chemical parameters, with residual sugar showing the highest coefficient of variation, suggesting diverse wine styles in the dataset.

Missing Values: Analysis confirmed no missing values in the dataset, ensuring complete case analysis for all statistical procedures.

Listing 5.3: Missing Values Analysis

```
# Check for missing values
cat("\n--- Missing Values Analysis ---\n")
missing_values <- sapply(wine_data, function(x) sum(is.na(x)))
```

```
print(missing_values)

if(sum(missing_values) == 0) {
        cat("No missing values found in the dataset.\n")
} else {
        cat("Missing values detected. See above for details.\n")
}
```

5.2.2 Distribution Analysis and Skewness

Listing 5.4: Skewness Analysis and Visualization

```
# Loop through numeric variables for histograms and skewness
skewness_results <- data.frame(
Variable = names(numeric_vars),
Skewness = numeric(ncol(numeric_vars)),
Interpretation = character(ncol(numeric_vars)),
stringsAsFactors = FALSE
)

for(i in 1:ncol(numeric_vars)) {
        var_name <- names(numeric_vars)[i]
        var_data <- numeric_vars[, i]

        # Calculate skewness
        skew_val <- skew(var_data, na.rm = TRUE)
        skewness_results$Skewness[i] <- skew_val

        # Interpret skewness
        if(abs(skew_val) < 0.5) {
                skewness_results$Interpretation[i] <- "Approxim
        } else if(skew_val >= 0.5) {
                skewness_results$Interpretation[i] <- "Right-ske
        } else {
                skewness_results$Interpretation[i] <- "Left-skew
        }

        # Create histogram
        hist(var_data,
        main = paste("Histogram of", var_name),
        xlab = var_name,
        col = "lightblue",
        border = "black")
}
```

[Insert skewness analysis table from R output]
Skewness analysis reveals important distributional characteristics:

- **Right-skewed variables** (skewness > 0.5): [List variables] suggest the presence of wines with elevated levels of these compounds
- **Left-skewed variables** (skewness $<$ -0.5): [List variables] indicate few wines with very low levels
- **Approximately symmetric variables** ($|\text{skewness}| < 0.5$): [List variables] follow roughly normal distributions

5.2.3 Outlier Detection

Listing 5.5: Outlier Detection using Boxplots

```
# Create boxplots for key variables
par(mfrow = c(2, 3))
key_vars <- c("fixed.acidity", "volatile.acidity", "alcohol",
"quality", "pH", "sulphates")

for(var in key_vars) {
        if(var %in% names(wine_data)) {
                boxplot(wine_data[[var]],
                main = paste("Boxplot of", var),
                ylab = var,
                col = "lightgreen")
        }
}
par(mfrow = c(1, 1))
```

Boxplot analysis identified potential outliers in several variables:
[Insert interpretation of boxplot results]
These outliers may represent wines with exceptional characteristics or measurement errors, but were retained in the analysis as they may contain valuable information about wine diversity.

5.2.4 Categorical Variable Distributions

Listing 5.6: Frequency Distributions for Categorical Variables

```
# Frequency distributions for categorical variables
cat("\n--- Frequency Distributions for Categorical Variables ---\n"
cat("Wine Variety Distribution:\n")
variety_table <- table(wine_data$variety)
print(variety_table)
```

```
print(prop.table(variety_table))

cat("\nQuality Distribution:\n")
quality_table <- table(wine_data$quality)
print(quality_table)
print(prop.table(quality_table))
```

Wine Variety Distribution:

- Red wines: [X] observations ([X]%)
- White wines: [X] observations ([X]%)

Quality Rating Distribution: *[Insert quality distribution table and interpretation]*

5.3 Hypothesis Testing

5.3.1 Research Question: Alcohol Content Comparison

Objective: Determine whether red and white wines differ significantly in alcohol content.

Listing 5.7: Alcohol Content Analysis by Wine Type

```
# Separate alcohol content by wine variety
red_alcohol <- wine_data$alcohol[wine_data$variety ==
    "red"]
white_alcohol <- wine_data$alcohol[wine_data$variety
   == "white"]

cat("--- Descriptive Statistics by Wine Type ---\n")
cat("Red wine alcohol content:\n")
cat("Mean:", mean(red_alcohol, na.rm = TRUE), "\n")
cat("SD:", sd(red_alcohol, na.rm = TRUE), "\n")
cat("N:", length(red_alcohol), "\n\n")

cat("White wine alcohol content:\n")
cat("Mean:", mean(white_alcohol, na.rm = TRUE), "\n")
cat("SD:", sd(white_alcohol, na.rm = TRUE), "\n")
cat("N:", length(white_alcohol), "\n\n")
```

5.3.2 Assumption Checking

Listing 5.8: T-Test Assumption Testing

```
# Check t-test assumptions
cat("--- Checking T-Test Assumptions ---\n")

# 1. Normality check using Shapiro-Wilk test
cat("1. Normality Tests:\n")
shapiro_red <- shapiro.test(red_alcohol)
shapiro_white <- shapiro.test(white_alcohol)

cat("Red wine alcohol normality (Shapiro-Wilk): p =",
shapiro_red$p.value, "\n")
cat("White wine alcohol normality (Shapiro-Wilk): p =
    ",
shapiro_white$p.value, "\n")

# 2. Equal variances test (Levene's test)
cat("\n2. Equal Variances Test (Levene):\n")
levene_test <- leveneTest(alcohol ~ variety, data =
    wine_data)
print(levene_test)

# 3. Independence assumption (addressed in
    interpretation)
cat("\n3. Independence: Assumed based on random
    sampling design\n")
```

Normality Assessment:

- Shapiro-Wilk test for red wines: p = [X.XXX]
- Shapiro-Wilk test for white wines: p = [X.XXX]

Equal Variances Assessment:

- Levene's test: F = [X.XX], p = [X.XXX]

Independence: Assumed based on sampling methodology

5.3.3 T-Test Results

Listing 5.9: Two-Sample T-Test

```
# Conduct t-test
cat("\n--- Two-Sample T-Test Results ---\n")

# Use Welch t-test (unequal variances) as default
```

```
t_test_result <- t.test(red_alcohol, white_alcohol,
    var.equal = FALSE)
print(t_test_result)

# Effect size (Cohen's d)
pooled_sd <- sqrt(((length(red_alcohol)-1)*var(red_
    alcohol) +
(length(white_alcohol)-1)*var(white_alcohol)) /
(length(red_alcohol) + length(white_alcohol) - 2))
cohens_d <- (mean(red_alcohol) - mean(white_alcohol))
     / pooled_sd

cat("\nEffect Size (Cohen's d):", cohens_d, "\n")
```

[Insert t-test output from R]

Statistical Decision: [Based on p-value, state whether to reject or fail to reject H_0]

Effect Size: Cohen's d = [X.XX], indicating [small/medium/large] effect size.

Interpretation: [Provide practical interpretation of the results]

5.4 Linear Regression Analysis

5.4.1 Model Specification

The linear regression model predicts wine quality using all available chemical and sensory variables for red wines only:

$$\text{Quality} = \beta_0 + \beta_1(\text{Fixed Acidity}) + \beta_2(\text{Volatile Acidity}) + \ldots + \beta_{11}(\text{Alcohol}) + \varepsilon \quad (5.1)$$

Listing 5.10: Linear Regression Model Setup

```
# Filter for red wines only
red_wines <- wine_data[wine_data$variety == "red", ]
cat("Number of red wine observations:", nrow(red_wines), "\n")

# Select chemical and sensory variables
predictor_vars <- c("fixed.acidity", "volatile.acidity", "citric
"residual.sugar", "chlorides", "free.sulfur.dioxide",
"total.sulfur.dioxide", "density", "pH", "sulphates", "alcohol")

# Create regression model
cat("\n--- Multiple Linear Regression Model ---\n")
regression_formula <- as.formula(paste("quality ~", paste(predic
wine_lm <- lm(regression_formula, data = red_wines)
```

```
# Display regression results
summary(wine_lm)
```

5.4.2 Regression Results

[Insert regression summary output]

Model Fit Statistics:

- R^2 = [X.XXX]: [X]% of variance in quality explained
- Adjusted R^2 = [X.XXX]: Accounts for number of predictors
- F-statistic = [X.XX], $p < 0.001$: Model is statistically significant

Significant Predictors ($\alpha = 0.05$): *[List significant variables with coefficients and interpretations]*

5.4.3 Regression Diagnostics

Listing 5.11: Regression Diagnostics

```
cat("\n--- Regression Diagnostics ---\n")

# 1. Linearity Check - Residuals vs Fitted Plot
cat("1. Linearity Assessment:\n")
plot(fitted(wine_lm), resid(wine_lm),
main = "Residuals vs Fitted Values",
xlab = "Fitted Values", ylab = "Residuals")
abline(h = 0, col = "red", lty = 2)

# RESET test for linearity
reset_test <- resettest(wine_lm, power = 2:3, type = "fitted")
cat("RESET Test for Linearity: p=", reset_test$p.value, "\n")

# 2. Normality of Residuals
cat("\n2. Normality of Residuals:\n")
shapiro_resid <- shapiro.test(resid(wine_lm))
cat("Shapiro-Wilk test on residuals: p=", shapiro_resid$p.value, "

# 3. Homoscedasticity
cat("\n3. Homoscedasticity Tests:\n")
bp_test <- bptest(wine_lm)
cat("Breusch-Pagan test: p=", bp_test$p.value, "\n")
```

```
# 4. Multicollinearity Check
cat("\n4. Multicollinearity Assessment:\n")
vif_values <- vif(wine_lm)
print(vif_values)

# 5. Autocorrelation Check
cat("\n5. Autocorrelation Test:\n")
dw_test <- dwtest(wine_lm)
cat("Durbin-Watson test: p =", dw_test$p.value, "\n")
```

Linearity Assessment:

- RESET test: p = [X.XXX]
- Residuals vs. Fitted plot interpretation: [Description]

Normality of Residuals:

- Shapiro-Wilk test on residuals: p = [X.XXX]
- Q-Q plot assessment: [Description]

Homoscedasticity:

- Breusch-Pagan test: p = [X.XXX]

Multicollinearity:

- VIF values: [List any values > 10 and interpretation]

Autocorrelation:

- Durbin-Watson test: p = [X.XXX]

Assumption Summary: [Overall assessment of regression assumptions and any violations]

5.5 Classification Methods

5.5.1 Good vs. Bad Wine Classification

Classification Scheme:

- Good wines: Quality ≥ 8
- Bad wines: Quality ≤ 4
- Medium wines: $5 \leq$ Quality ≤ 7 (excluded from analysis)

Listing 5.12: Binary Classification Setup

```
# Create binary classification: Good (>=8) vs Bad
    (<=4)
wine_data$quality_binary <- ifelse(wine_data$quality
    >= 8, "Good",
ifelse(wine_data$quality <= 4,
"Bad", "Medium"))

# Filter for only Good and Bad wines (exclude Medium)
classification_data <- wine_data[wine_data$quality_
    binary %in%
c("Good", "Bad"), ]
classification_data$quality_binary <-
factor(classification_data$quality_binary,
levels = c("Bad", "Good"))

cat("Classification Distribution:\n")
print(table(classification_data$quality_binary))

# Prepare predictor variables
predictor_formula <- as.formula(paste("quality_binary
    ~",
paste(predictor_vars,
collapse = " + ")))
```

Sample Distribution:

- Good wines: [X] observations
- Bad wines: [X] observations

5.5.2 Logistic Regression Model

Listing 5.13: Logistic Regression for Quality Classification

```
# Logistic Regression Model
cat("\n--- Logistic Regression Model ---\n")
quality_glm <- glm(predictor_formula, data =
    classification_data,
family = binomial)
summary(quality_glm)

# Model fit statistics
cat("\nModel Fit Statistics:\n")
cat("AIC:", AIC(quality_glm), "\n")
```

```
cat("Null Deviance:", quality_glm$null.deviance, "\n"
    )
cat("Residual Deviance:", quality_glm$deviance, "\n")
```

[Insert logistic regression summary]
Model Fit:

- AIC: [XXX.XX]
- Null Deviance: [XXX.XX]
- Residual Deviance: [XXX.XX]

5.5.3 Classification Performance

Listing 5.14: Classification Performance Evaluation

```
# Predictions and Classification Performance
predicted_probs <- predict(quality_glm, type = "
    response")
predicted_class <- ifelse(predicted_probs > 0.5, "
    Good", "Bad")

# Confusion Matrix
conf_matrix <- table(Actual = classification_data$
    quality_binary,
Predicted = predicted_class)
cat("\nConfusion Matrix:\n")
print(conf_matrix)

# Calculate performance metrics
accuracy <- sum(diag(conf_matrix)) / sum(conf_matrix)
sensitivity <- conf_matrix[2,2] / sum(conf_matrix
    [2,])
specificity <- conf_matrix[1,1] / sum(conf_matrix
    [1,])

cat("\nClassification Performance:\n")
cat("Accuracy:", round(accuracy, 4), "\n")
cat("Sensitivity (True Positive Rate):", round(
    sensitivity, 4), "\n")
cat("Specificity (True Negative Rate):", round(
    specificity, 4), "\n")
```

[Insert confusion matrix]
Performance Metrics:

- Accuracy: [X.XXX]
- Sensitivity (True Positive Rate): [X.XXX]
- Specificity (True Negative Rate): [X.XXX]

5.5.4 Wine Type Prediction

Methodology: Train-validation split (70-30) for model development and evaluation.

Listing 5.15: Wine Type Prediction with Validation

```
# Convert variety to binary (0/1) as required
wine_data$variety_binary <- ifelse(wine_data$variety
   == "red", 1, 0)

# Split data into training and validation sets (70/30
    split)
set.seed(123)  # For reproducibility
train_indices <- sample(nrow(wine_data), size = 0.7 *
    nrow(wine_data))
train_data <- wine_data[train_indices, ]
validation_data <- wine_data[-train_indices, ]

cat("Training set size:", nrow(train_data), "\n")
cat("Validation set size:", nrow(validation_data), "\
   n")

# Build logistic regression model on training data
variety_formula <- as.formula(paste("variety_binary ~
   ",
paste(predictor_vars,
collapse = " + ")))
variety_glm <- glm(variety_formula, data = train_data
   ,
family = binomial)

cat("\n--- Wine Type Prediction Model Summary ---\n")
summary(variety_glm)

# Predictions on validation set
validation_probs <- predict(variety_glm, newdata =
   validation_data,
type = "response")
validation_pred <- ifelse(validation_probs > 0.5, 1,
   0)
```

```
# Confusion Matrix on validation set
validation_conf <- table(Actual = validation_data$
    variety_binary,
Predicted = validation_pred)
cat("\nValidation Set Confusion Matrix:\n")
print(validation_conf)
```

Training Set Performance: *[Insert training model summary]*
Validation Set Results: *[Insert validation confusion matrix and metrics]*

Listing 5.16: ROC Analysis

```
# Performance metrics on validation set
val_accuracy <- sum(diag(validation_conf)) / sum(
    validation_conf)
val_sensitivity <- validation_conf[2,2] / sum(
    validation_conf[2,])
val_specificity <- validation_conf[1,1] / sum(
    validation_conf[1,])

cat("\nValidation Set Performance:\n")
cat("Accuracy:", round(val_accuracy, 4), "\n")
cat("Sensitivity:", round(val_sensitivity, 4), "\n")
cat("Specificity:", round(val_specificity, 4), "\n")

# ROC Curve and AUC
cat("\n--- ROC Analysis ---\n")
roc_obj <- roc(validation_data$variety_binary,
    validation_probs)
auc_value <- auc(roc_obj)

cat("AUC Value:", round(auc_value, 4), "\n")

# Plot ROC curve
plot(roc_obj, main = "ROC Curve - Wine Type
    Prediction",
col = "blue", lwd = 2)
abline(a = 0, b = 1, lty = 2, col = "red")
legend("bottomright", paste("AUC=", round(auc_value,
    4)),
col = "blue", lwd = 2)
```

ROC Analysis:

- AUC = [X.XXX]

- Interpretation: [Outstanding/Excellent/Acceptable/Poor] classification performance

5.6 Factor Analysis

5.6.1 Suitability Assessment

Listing 5.17: Factor Analysis Preparation

```
# Prepare data for factor analysis (chemical and
    sensory variables only)
factor_data <- wine_data[, predictor_vars]

# Check for missing values
cat("Missing values in factor analysis data:\n")
print(sapply(factor_data, function(x) sum(is.na(x))))

# Correlation matrix assessment
cat("\n--- Correlation Matrix Suitability ---\n")
correlation_matrix <- cor(factor_data, use = "
    complete.obs")
```

Listing 5.18: KMO and Bartlett Tests

```
# Kaiser-Meyer-Olkin (KMO) Test
kmo_result <- KMOS(factor_data)
kmo_overall <- kmo_result$KMO
msa_values <- kmo_result$MSA

cat("Overall KMO value:", round(kmo_overall, 4), "\n"
    )
if(kmo_overall >= 0.8) {
        cat("KMO assessment: Excellent for factor
            analysis\n")
} else if(kmo_overall >= 0.7) {
        cat("KMO assessment: Good for factor analysis
            \n")
} else if(kmo_overall >= 0.6) {
        cat("KMO assessment: Adequate for factor
            analysis\n")
} else if(kmo_overall >= 0.5) {
        cat("KMO assessment: Poor but acceptable\n")
} else {
        cat("KMO assessment: Unacceptable for factor
            analysis\n")
```

```
}

# Bartlett's Test of Sphericity
bartlett_result <- cortest.bartlett(correlation_
    matrix, n = nrow(factor_data))
cat("\nBartlett's Test of Sphericity: p =", bartlett_
    result$p.value, "\n")

if(bartlett_result$p.value < 0.05) {
        cat("Bartlett's test: Correlations exist (
            suitable for factor analysis)\n")
} else {
        cat("Bartlett's test: No significant
            correlations (not suitable)\n")
}
```

Kaiser-Meyer-Olkin (KMO) Test:

- Overall KMO = [X.XXX]
- Assessment: [Excellent/Good/Adequate/Poor/Unacceptable] for factor analysis

Bartlett's Test of Sphericity:

- p-value < 0.001: Correlations exist, suitable for factor analysis

Measure of Sampling Adequacy (MSA): *[Insert MSA values for individual variables]*

5.6.2 Factor Extraction

Listing 5.19: Determining Number of Factors

```
# MSA for individual variables
cat("\nMeasure of Sampling Adequacy (MSA) for
    individual variables:\n")
msa_df <- data.frame(Variable = names(msa_values),
MSA = round(msa_values, 4))
print(msa_df)

# Remove variables with MSA < 0.5 if any
low_msa_vars <- names(msa_values)[msa_values < 0.5]
if(length(low_msa_vars) > 0) {
        cat("\nVariables with MSA < 0.5 (consider
            removal):", low_msa_vars, "\n")
        factor_data_reduced <- factor_data[, !names(
            factor_data) %in% low_msa_vars]
```

```
} else {
        cat("\nAll variables have acceptable MSA (>=
            0.5)\n")
        factor_data_reduced <- factor_data
}

# Determine number of factors
cat("\n--- Determining Number of Factors ---\n")

# Eigenvalues
eigenvalues <- eigen(cor(factor_data_reduced))$values
cat("Eigenvalues:\n")
for(i in 1:length(eigenvalues)) {
        cat("Factor", i, ":", round(eigenvalues[i],
            4), "\n")
}

# Kaiser criterion (eigenvalues > 1)
n_factors_kaiser <- sum(eigenvalues > 1)
cat("\nKaiser criterion (eigenvalues > 1):", n_
    factors_kaiser, "factors\n")

# Scree plot
plot(1:length(eigenvalues), eigenvalues, type = "b",
main = "Scree Plot", xlab = "Factor Number", ylab = "
    Eigenvalue")
abline(h = 1, col = "red", lty = 2)
```

Eigenvalue Analysis: *[Insert eigenvalues and Kaiser criterion results]*
Scree Plot Interpretation: [Description of scree plot and elbow criterion]

5.6.3 Factor Analysis Results

Listing 5.20: Factor Analysis Implementation

```
# Conduct factor analysis
cat("\n--- Factor Analysis Results ---\n")

# Try different numbers of factors
for(nf in 1:min(4, n_factors_kaiser)) {
        cat("\n=== Factor Analysis with", nf, "factor
            (s) ===\n")

        fa_result <- principal(factor_data_reduced,
            nfactors = nf,
```

```
                rotate = "varimax")

                # Factor loadings
                cat("Factor Loadings (>0.4 shown):\n")
                print(fa_result, cut = 0.4, sort = TRUE)

                # Variance explained
                variance_explained <- fa_result$values[1:nf]
                total_variance <- sum(variance_explained)
                proportion_variance <- total_variance / ncol(
                    factor_data_reduced)

                cat("\nVariance Explained:\n")
                cat("Total eigenvalues for", nf, "factors:",
                round(total_variance, 4), "\n")
                cat("Proportion of variance explained:",
                round(proportion_variance, 4), "\n")
                cat("Percentage of variance explained:",
                round(proportion_variance * 100, 2), "%\n")
        }
```

[X]-Factor Solution: *[Insert factor loadings table with >0.4 loadings]*
Variance Explained:

- Total variance explained: [XX.X]%
- Interpretation of factors: [Describe what each factor represents]

Factor Interpretation:

- Factor 1: [Description based on loadings]
- Factor 2: [Description based on loadings]

e for additional factors

6 Conclusion

6.1 Summary

This comprehensive analysis of Portuguese wine data has provided valuable insights into the relationships between chemical composition and wine quality characteristics. The key findings from each analytical component are summarized below:

Exploratory Data Analysis revealed diverse chemical profiles across the wine dataset, with most variables showing reasonable distributions suitable for statistical analysis. Skewness analysis identified several right-skewed variables, indicating the presence of wines with elevated levels of certain compounds.

Hypothesis Testing for alcohol content differences between red and white wines [resulted in rejection/failure to reject of the null hypothesis], [indicating significant/no significant] differences between wine types. This finding has implications for wine classification and production understanding.

Linear Regression Analysis of red wine quality demonstrated that [X]% of quality variance can be explained by chemical variables. Significant predictors included [list key variables], suggesting these compounds are critical for quality assessment. Regression diagnostics indicated [summary of assumption compliance].

Classification Analysis showed [moderate/high/low] performance in distinguishing good from bad wines, with accuracy of [X]%. The wine type prediction model achieved excellent performance (AUC = [X.XX]), demonstrating that chemical composition strongly distinguishes red from white wines.

Factor Analysis successfully reduced the dimensionality of chemical variables to [X] underlying factors, explaining [XX]% of total variance. These factors appear to represent [brief description of factor interpretation].

6.2 Outlook

6.2.1 Theoretical Implications

The results contribute to the understanding of wine quality assessment from a data science perspective. The successful application of multiple statistical methods demonstrates the value of quantitative approaches in enological research. The factor structure identified in chemical composition data provides insights into the underlying dimensions of wine chemistry that could inform future research directions.

6.2.2 Practical Applications

Wine Industry Applications:

- Quality control systems could implement the developed models for objective quality assessment
- Chemical analysis protocols could focus on the most predictive variables identified
- Classification models could assist in automated wine categorization

Future Research Directions:

- Extension to larger datasets with diverse wine regions and grape varieties
- Integration of spectroscopic data with traditional chemical analysis
- Development of real-time quality monitoring systems for wine production
- Investigation of temporal changes in wine chemistry and quality relationships

6.2.3 Methodological Considerations

This study demonstrates the successful application of statistical methods taught in Introduction to Data Science to a real-world problem. The combination of exploratory analysis, hypothesis testing, regression, classification, and dimensionality reduction provides a comprehensive analytical framework that could be adapted to other food science applications.

Limitations:

- Dataset limited to Portuguese wines, potentially affecting generalizability
- Quality ratings based on expert panels, which may introduce subjective bias
- Chemical analysis limited to standard parameters, excluding emerging quality indicators

Recommendations for Future Studies:

- Expand dataset to include international wine varieties
- Investigate machine learning methods beyond logistic regression
- Incorporate consumer preference data alongside expert quality ratings
- Develop ensemble models combining multiple analytical approaches

The integration of traditional statistical methods with modern data science techniques demonstrates significant potential for advancing wine quality research and practical applications in the wine industry.