Predicting Wine Quality with Chemical and Physical Composition

STA 135 | Feini Pek, Hebi Wang, Fanling Liu, Jieying Ma | December 2023

This report examines the correlation between chemical attributes and wine types and quality assessments in 600 Portuguese wines, aiming to understand how factors like acidity, sugar content, sulfur levels, and alcohol content influence wine quality ratings. Three assessors rated each wine on a 10-point scale, with the median score determining the final rating. The explores differences between red and white wines using PCA and K-means clustering, with K-means showing a 90.25% accuracy in wine category prediction after dimensionality reduction. Linear Discriminant Analysis (LDA) further demonstrated high predictive accuracy for classifying wine types, with a low mislabeling rate of 0.3%. The report also investigates the distinguishing features of the highest-quality wines. Volatile acidity, free sulfur dioxide, and alcohol levels were found to be significant influencers of wine quality, with a multinomial logistic regression model achieving up to 65.8% accuracy in quality prediction. Furthermore, the LDA model showed a 21.6% misclassification rate for classifying varying wine qualities. A relatively high error rate indicates a need for a cautious interpretation of predicting wine quality models.

Introduction

Wine is a product meant not only to be consumed and enjoyed but also marketed and sold. As such, factors like taste, quality, and uniqueness play an important role in determining a wine's appeal and value. Defining "quality" for wine can be complex given the many elements that influence sensory characteristics (Basalekou et al., 2023). This report seeks to examine contributing factors to wine quality by examining the relationship between various chemical measurements and quality ratings by three assessors. Specifically, it analyzes how factors like acidity, sugar content, sulfur levels, alcohol content, and content correlate with the quality assessment of 600 Portuguese wines. Assessors rated the overall quality of each wine on a 10-point scale, with the median score used as the final quality rating. We investigate how these chemical measurements relate to quality ratings for both red and white wines. The goal is to better understand how measurable traits like acidity influence perceived quality by human tasters. These insights can help guide winemakers in their efforts to improve wine quality and appeal.

Results

1. Summarize the data

In the given 13 variables, "quality" and "red" are grouping variables, and the rest are continuous. The variables have different ranges and variances, which means they need to be standardized (see Table 1 below). Figure 1, below, is reported the distributions of the variables. We can see outliers in chlorides and sulfates, which need to be removed. After removing outliers, Figure 2 shows the correlation between variables. The most correlated variables are free sulfur dioxide and total sulfur dioxide, and the least correlated variables are red and total sulfur dioxide. Figure 3 shows 4 pairs of variables that correlate greater than 0.5. The PCA result in Figure 4 shows that the first two principle components can explain 0.548 of the variance.

2. Multivariate Normality

In Figure 1, presented below, the findings for Multivariate Normality are displayed. The marginal normal quantile-quantile (Q-Q) plots for Principal Component 1 (PC1) exhibit a slight right skew, whereas the Q-Q plots for Principal Component 2 (PC2) demonstrate a pronounced heavy tail on the right, potentially attributable to multiple outliers in that region. The overall chi-squared QQ plot illustrates a heavy tail on the right, where the quantiles obviously depart from normality. This deviation suggests a notable impact of right-side outliers on the observed non-normality.

3. Differences between red and white wines

3.1 Use PCA and K-means clustering to predict red and white wines.

First of all, we used one-way MANOVA to test whether there is a significant difference between red wine and white wine. The p-value in Table 1 shows there is a difference between the two categories of wine. Explore the appropriate number of principal components. Figure 2 shows that the first two principal components explained most of the cumulative proportion of variance. Figure 3 is the scatter plot of PC1 and PC2. Using K-means clusters to predict the wine category. It divides the dots in the scatter plot of PC1 and PC2 into two categories, with red dots representing red wine and black dots representing white wine. Compare the real value with the predicted value. Figure 5 stands for the real category of wine. Compare Figure 4 with Figure 5, only partial points at the junction of the two clusters were predicted incorrectly. The error rate is around 9.75%. This means that after dimensionality reduction of the data by principal component analysis, the accuracy of wine category prediction by the k-means clustering method is 90.25%.

3.2 Use LDA to predict red and white wines.

The wine data set was split into training and testing sets. The coefficients of the linear discriminants are detailed in the Appendix. Importantly, the Linear Discriminant Analysis (LDA) model constructed demonstrates considerable predictive proficiency when applied to the testing dataset, evidenced by a remarkably low mislabeling rate of 0.0033. This performance instills confidence in the LDA model's capability to accurately classify red and white wines based on the specified 'x' variables related to wine characteristics.

3.3 Use K-means clustering to predict red and white wines.

The K-means clustering method exhibits effective classification capabilities for distinguishing between red and white wines. With a mislabeling rate of 0.016, its accuracy is commendable, slightly less precise than that achieved through LDA classification. Nonetheless, the performance of K-means clustering is sufficiently robust, making it a viable approach for classifying red and white wines.

4. Differences between the highest quality wines and the others

4.1 Multinomial Logistic Regression

Recent research suggests that volatile acidity, free sulfur dioxide levels, and alcohol content are key factors influencing wine quality evaluation (Basalekou et al., 2023). We analyzed our dataset to investigate whether it supports this claim regarding the impact of these chemical measures on quality ratings. Visual inspection of boxplots (Figure 4.1.1) reveals potential differences in the distributions of these variables across the 3 quality rating categories (5, 6, and 7). To statistically test for significant differences, we ran Tukey's HSD test. The results indicate significant differences in volatile acidity between all rating groups ($p \approx 0.05$) and also significant differences in alcohol content ($p \approx 0$). However, there were no statistically significant differences in free sulfur dioxide among groups at $\alpha = 0.05$ (Figure 4.1.2). Given these initial findings, we fit a multinomial logistic regression model using quality rating as the outcome and volatile acidity and alcohol level as predictor variables. This model yields negative coefficients for volatile acidity and free sulfur dioxide and a positive coefficient for alcohol level. This implies that by holding everything equal, a higher concentration of volatile acidity and free sulfur dioxide may lower a wine's quality, whereas a higher alcohol content will increase the wine's quality. This model correctly predicted quality categories 63.3% of the time. The addition of free sulfur dioxide as a predictor improved the prediction accuracy to 65.8%. In summary, our analysis provides evidence that volatile acidity, free sulfur dioxide, and alcohol content have measurable influence over statistically modeling quality ratings in this wine data set. While not all factors showed significant variation across groups in univariate analysis, the multivariate regression results suggest they each contribute some signal as to wine quality perceptions.

4.2 Use LDA to distinguish between the highest quality wines and other quality grades.

The wine dataset was divided into training and testing subsets, with the coefficients for the linear discriminants provided in the Appendix, which shows that fixed acidity, density, and residual sugar carry the highest weight in discrimination (-0.96, 0.88, 0.70, respectively). The LDA model shows reasonable predictive accuracy when applied to the testing set, as indicated by a misclassification rate of 0.216. This rate, while not negligible, offers a degree of confidence in the LDA model's ability to classify wines of varying qualities based on the 'x' variables, which are attributes of wine. However, the presence of a 21.6% misclassification rate, coupled with potential variance across different samples, suggests a need for caution in relying on this model for precise predictions.

Discussion

By comparing the three predicted methods, we can see that the LDA model has the lowest error rate. The second one is to use only k-means clustering. It is indeed possible to predict wine categories by using PCA and k-means clustering at the same time, probably because they are inherently clustered, and the correct rate is also relatively ideal. However, its prediction accuracy is far inferior to the other two prediction methods. At the same time, clustering and principal components are no longer applicable in problem 5, they are only feasible for some problems.

To study how continuous measurement indicates differences between wine qualities, we utilized two statistical approaches: multinomial logistic regression and linear discriminant analysis (LDA). The multinomial regression model aimed to predict quality rating categories (5, 6, or 7) using volatile acidity, free sulfur dioxide, and alcohol content as predictor variables. Although boxplot inspection and the Tukey HSD test imply that there is no significant difference in free sulfur dioxide content between different wine qualities, including it as a predictor alongside volatile acidity and alcohol slightly improved model accuracy to 65.8%. Meanwhile, the LDA classified wines into two groups: high quality (rating 7) and lower quality (ratings 5 and 6). The LDA achieved a 78.4% accuracy rate in distinguishing these groups. We also observed that fixed acidity, density, residual sugar, and alcohol content have the biggest weight in LDA. In comparing the two techniques, LDA yielded higher predictive performance, though with a simpler two-category outcome. The drivers of quality perceptions differed as well; while both implicated alcohol, the LDA highlighted the additional roles of attributes such as fixed acidity and sugar level. In summary, both approaches provide supporting evidence that chemical aspects like acidity and alcohol are associated with sensory experience. However, wine appreciation includes many intangible factors as well. Future work could investigate how measures like sulfates or pH also influence quality ratings.

Conclusion

This study reveals a clear link between chemical attributes and wine quality in Portuguese wines. By employing PCA and K-means clustering, we effectively differentiated between red and white wines based on their chemical makeup. Linear Discriminant Analysis (LDA) proved to be a robust method for classifying wine types. Our findings also confirm the influence of volatile acidity, free sulfur dioxide, and alcohol levels on wine quality, with a logistic regression model predicting quality. However, the logistics and LDA model's misclassification rate for quality highlights the complexity of accurately predicting wine quality, emphasizing the intricate balance between chemical properties and subjective tasting experiences. These findings offer important insights for winemakers and consumers in understanding and classifying wines.

Table 1.1: Type of variables and descriptive statistics.

	Type	Mean	Median	SD
fixed.acidity	double	7.585	7.2	1.569
volatile.acidity	double	0.387	0.34	0.18
citric.acid	double	0.309	0.31	0.164
residual.sugar	double	4.436	2.4	4.144
chlorides	double	0.064	0.059	0.035
free.sulfur.dioxide	double	26.27	24	16.479
total.sulfur.dioxide	double	92.026	89	58.691
density	double	0.995	0.996	0.003
pH	double	3.249	3.25	0.161
sulphates	double	0.584	0.55	0.181
alcohol	double	10.613	10.5	1.208
quality	integer	6	6	0.817
red	integer	0.5	0.5	0.5

Figure 1.1: Variables distribution.

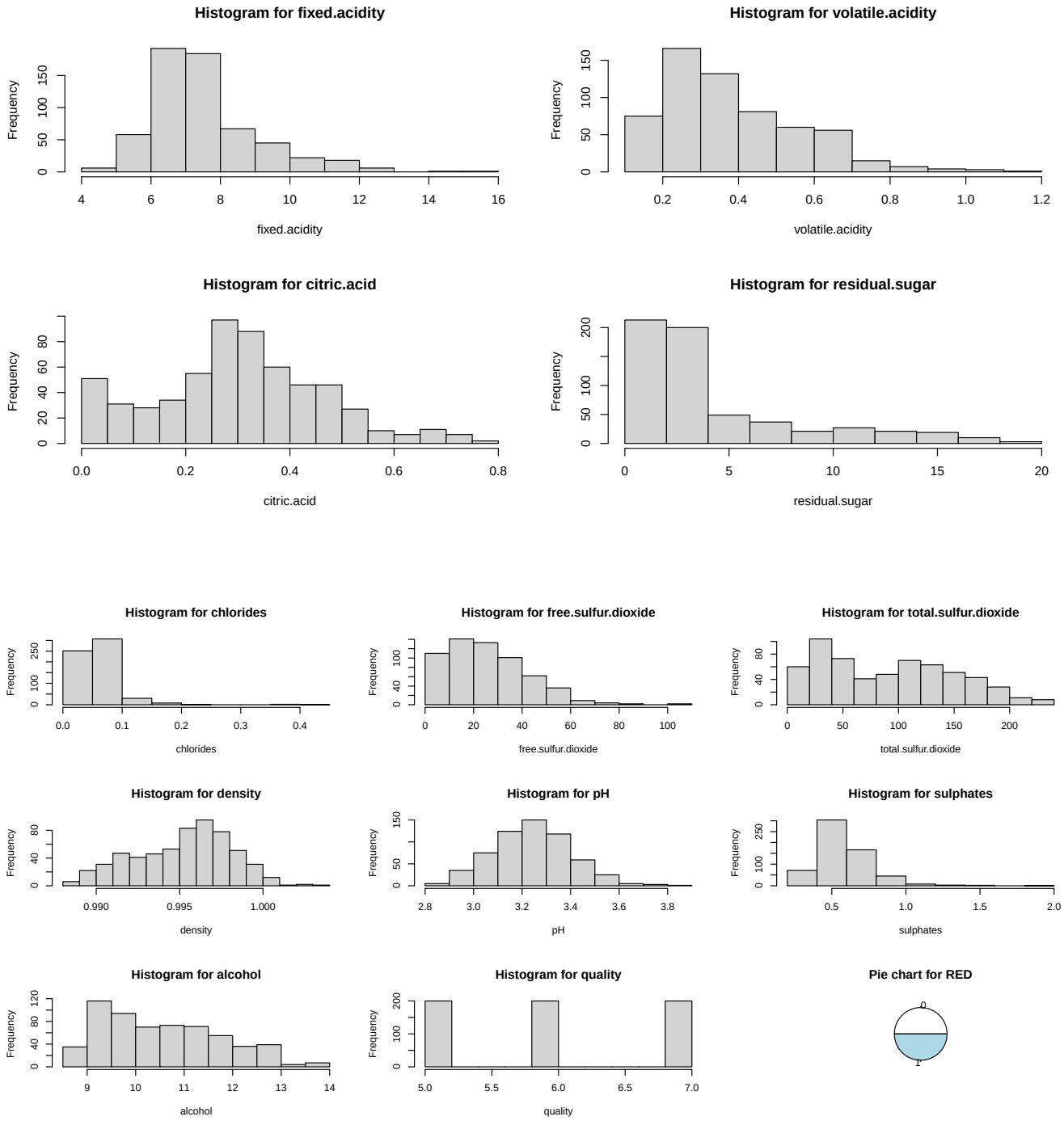


Figure 1.2: Correlation between variables.

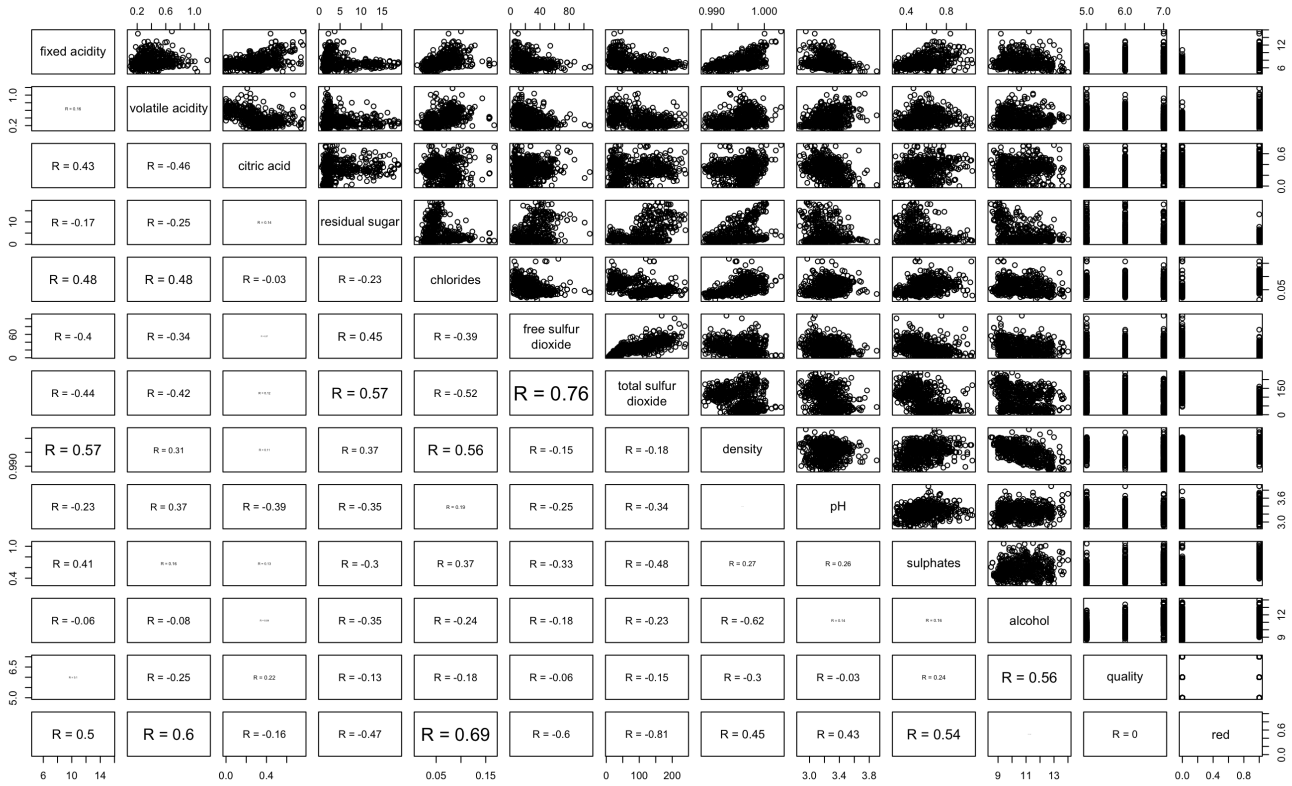


Figure 1.3: Highly correlatived variables.

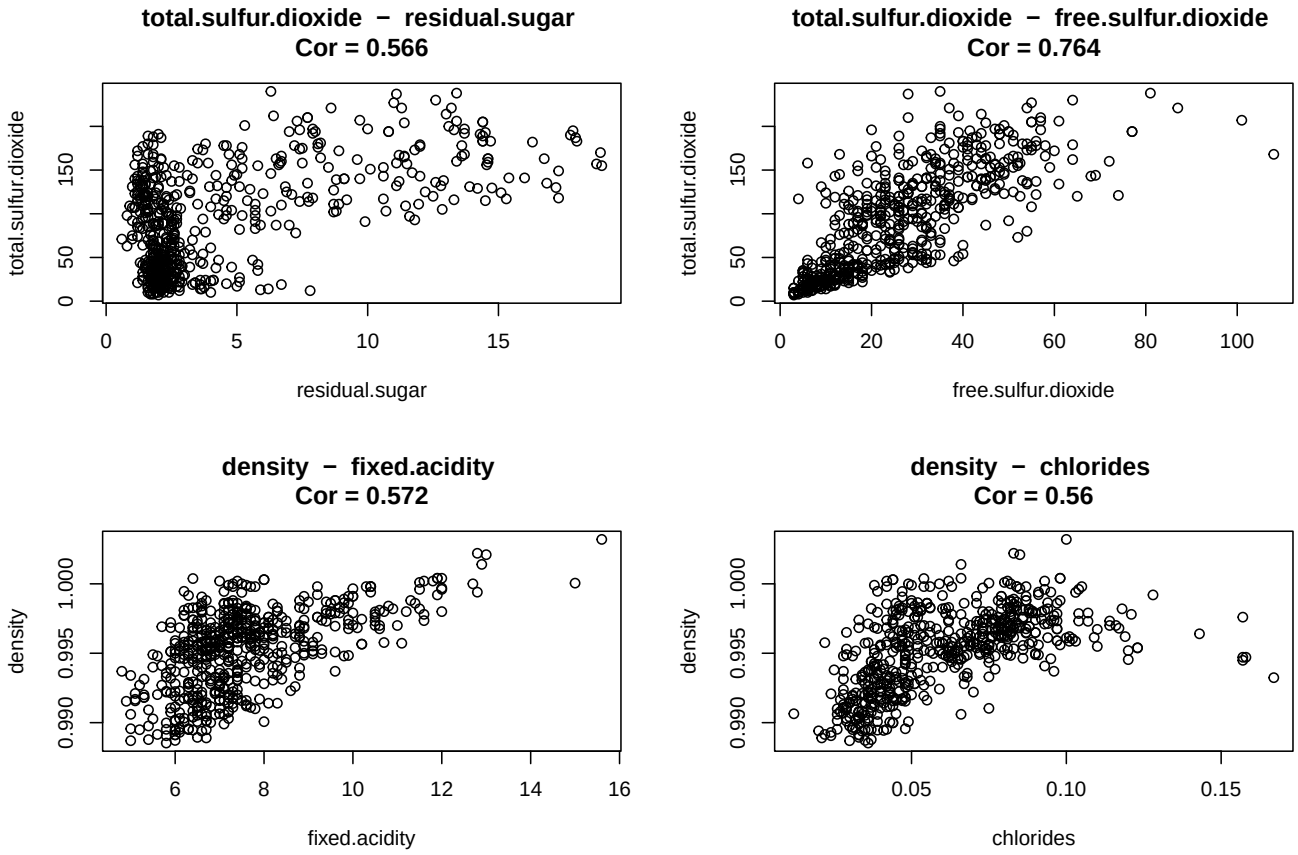
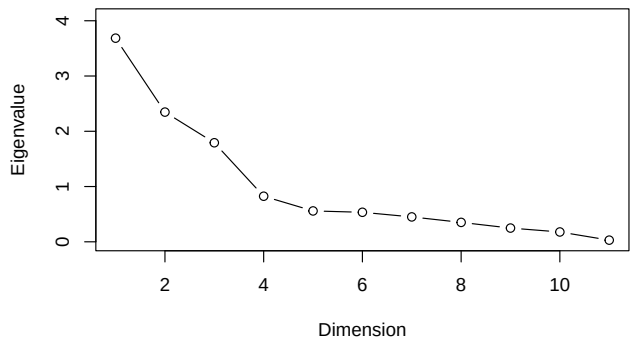
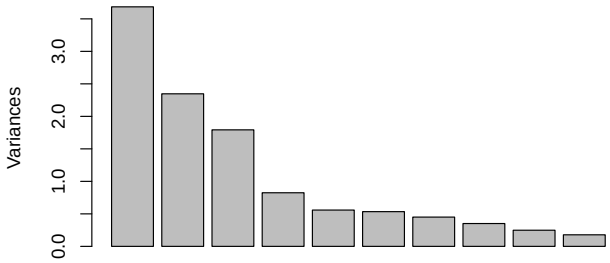
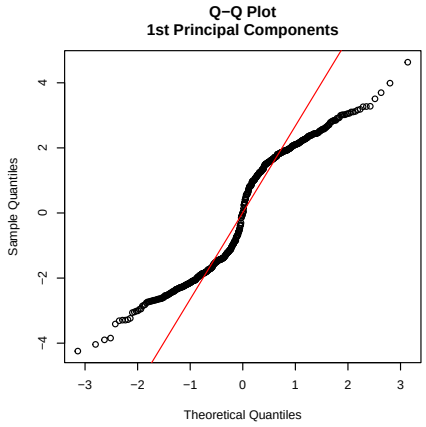


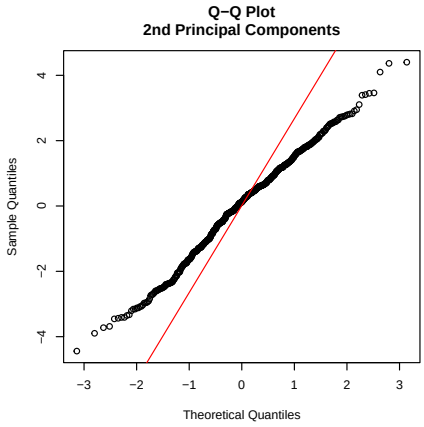
Figure 1.4: PCA Result.

```
## [1] 0.5483613
```









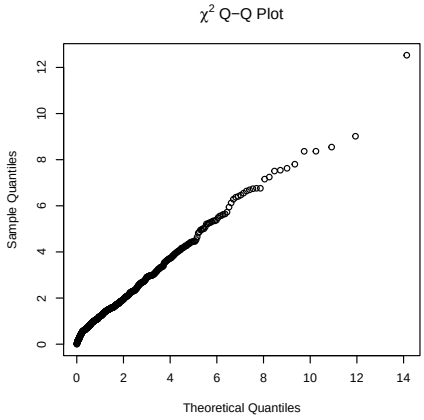


Figure 2.1: Results of Multivariate Normality.

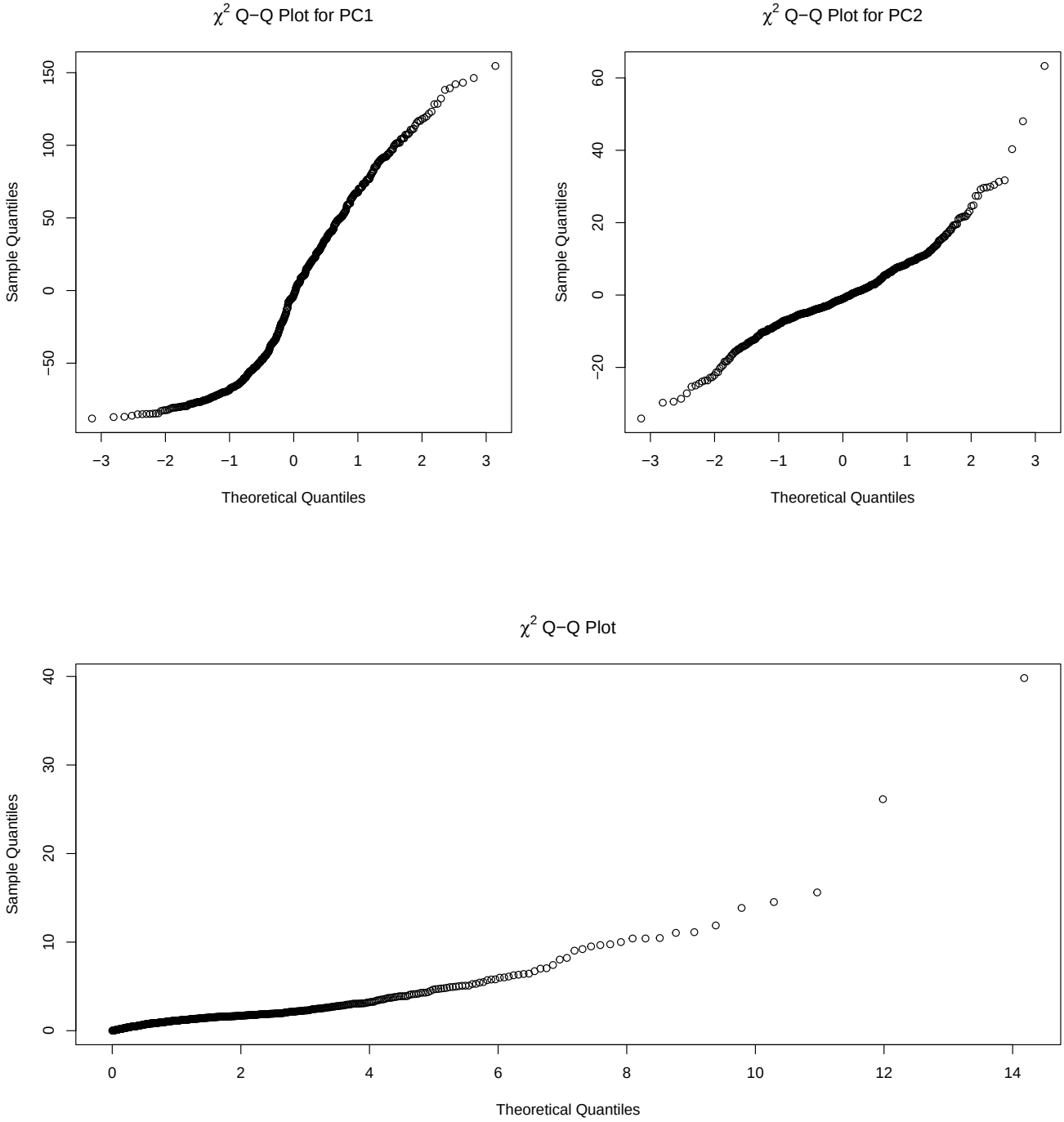


Table 3.1.1: Results from one-way MANOVA.

```
##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1           8.1            0.670        0.55            1.8     0.117
## 2           9.6            0.680        0.24            2.2     0.087
## 3           7.7            1.005        0.15            2.1     0.102
## 4           7.1            0.340        0.28            2.0     0.082
## 5           8.3            0.650        0.10            2.9     0.089
## 6           8.8            0.685        0.26            1.6     0.088
##   free.sulfur.dioxide total.sulfur.dioxide density   pH sulphates alcohol
## 1                  32                 141 0.99680 3.17      0.62     9.4
## 2                   5                  28 0.99880 3.14      0.60    10.2
## 3                  11                  32 0.99604 3.23      0.48    10.0
## 4                  31                  68 0.99694 3.45      0.48     9.4
## 5                  17                  40 0.99803 3.29      0.55     9.5
## 6                  16                  23 0.99694 3.32      0.47     9.4
##   quality red
## 1       5   1
## 2       5   1
## 3       5   1
## 4       5   1
## 5       5   1
## 6       5   1

## Analysis of Variance Table
##
##              Df   Wilks  approx F num Df den Df    Pr(>F)
## (Intercept)   1 0.00000 129775123     11    588 < 2.2e-16 ***
## red           1 0.12043       390     11    588 < 2.2e-16 ***
## Residuals   598
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 3.1.2: Results from scree plots of PCA.

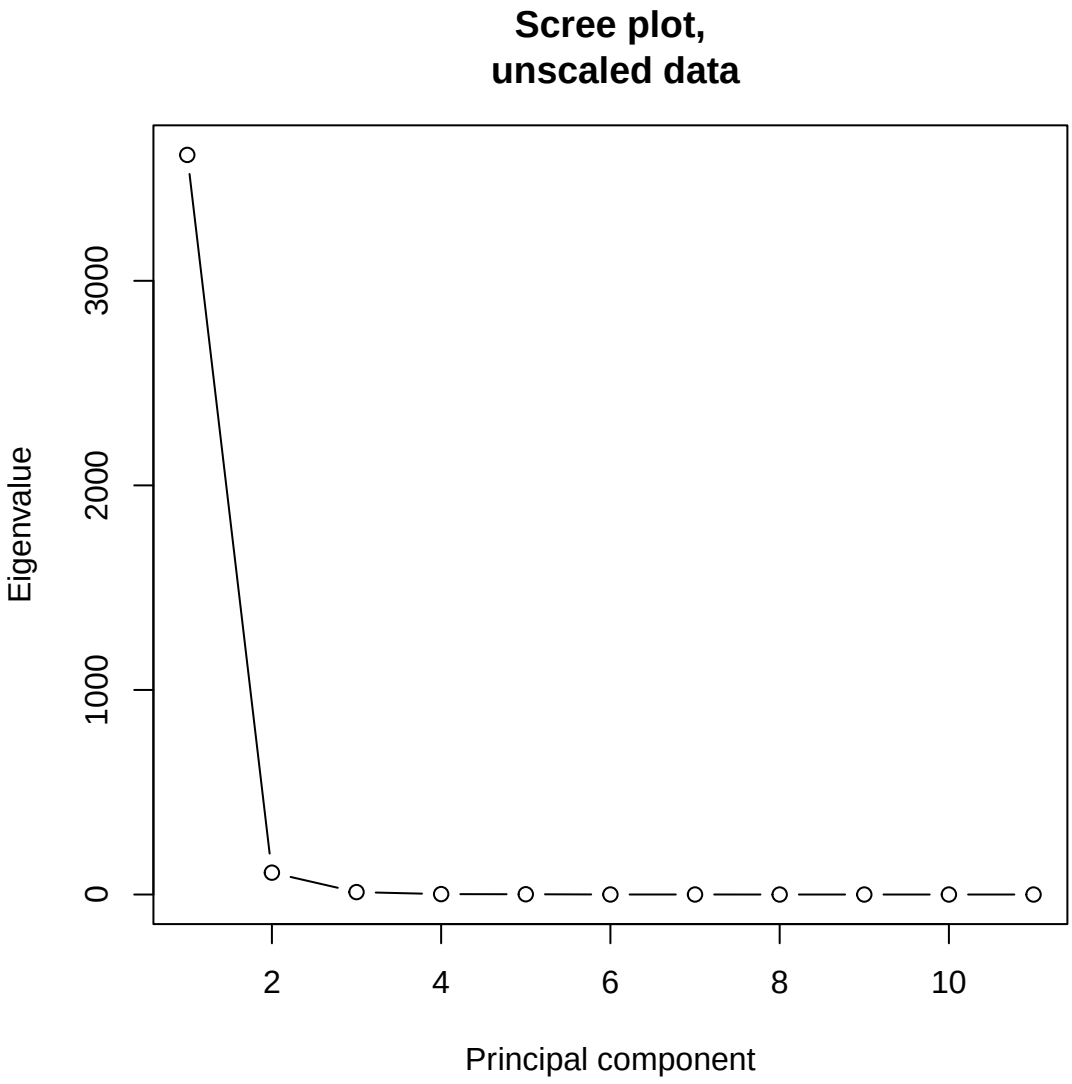


Figure 3.1.3: plot of Principal components.

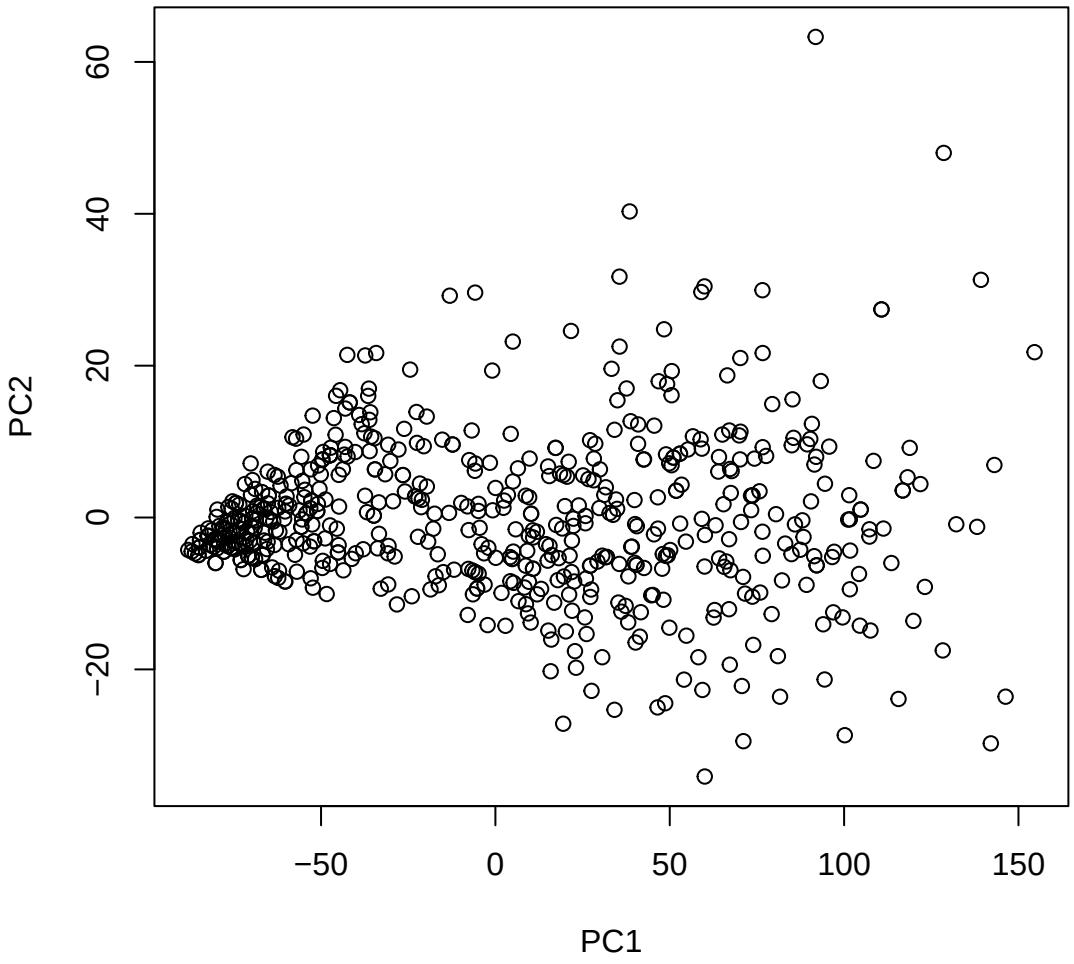


Figure 3.1.4: results from using K-means cluster to predict the category of wine.

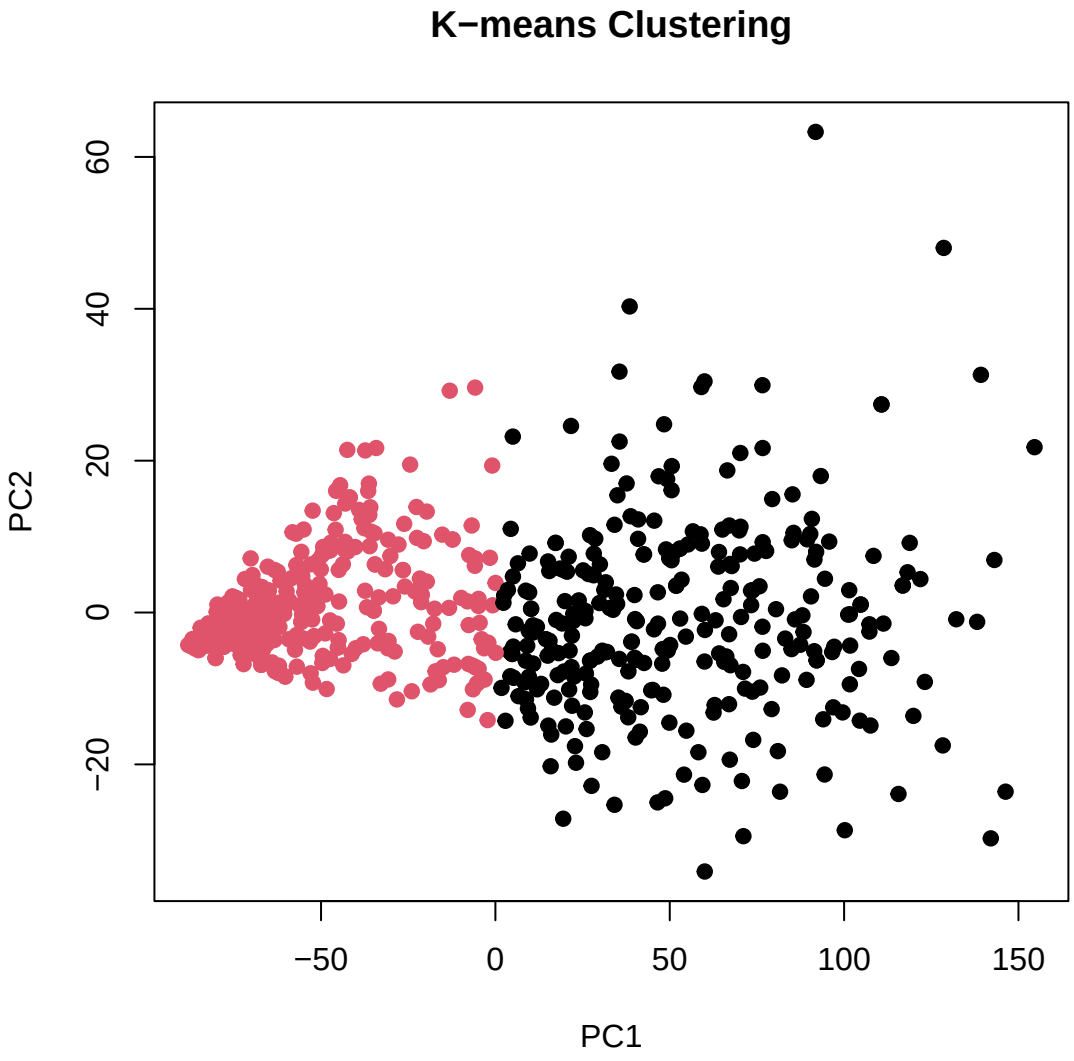


Figure 3.1.5: The real category of wine

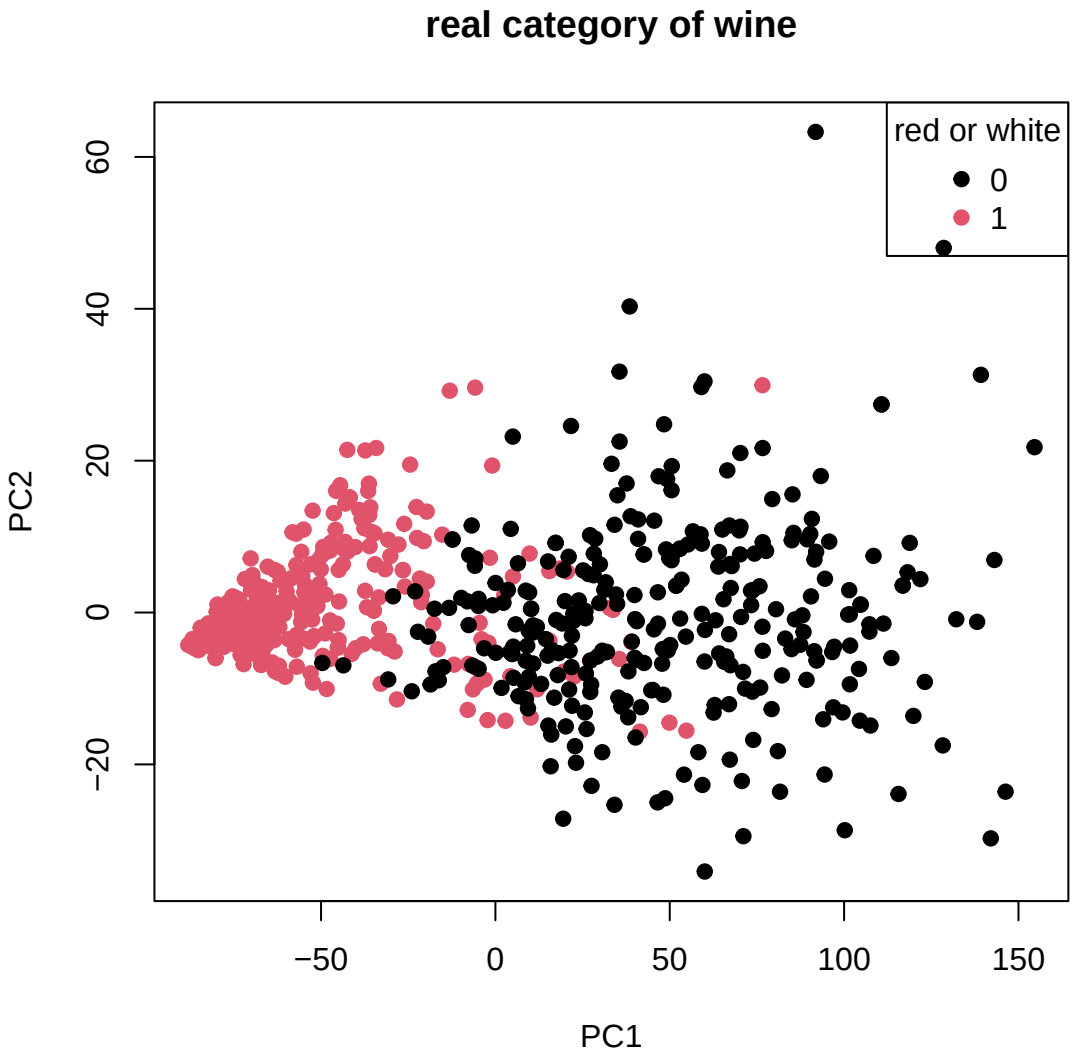
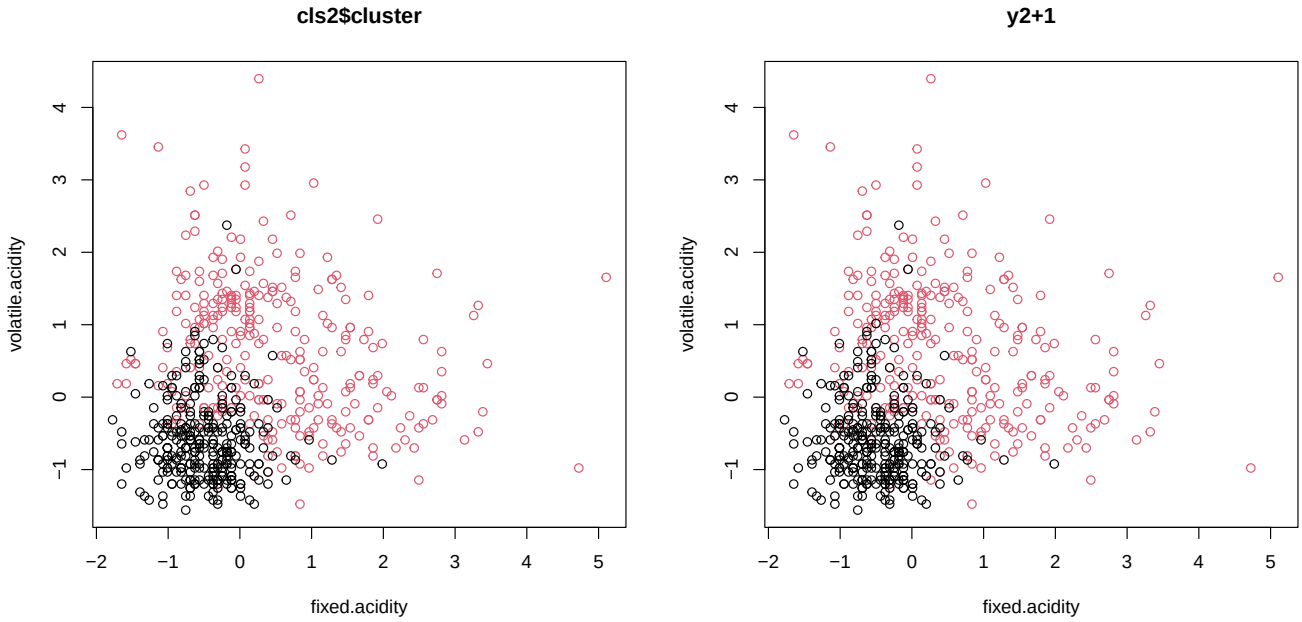


Table 3.2: Results of LDA in classifying red and white wine.

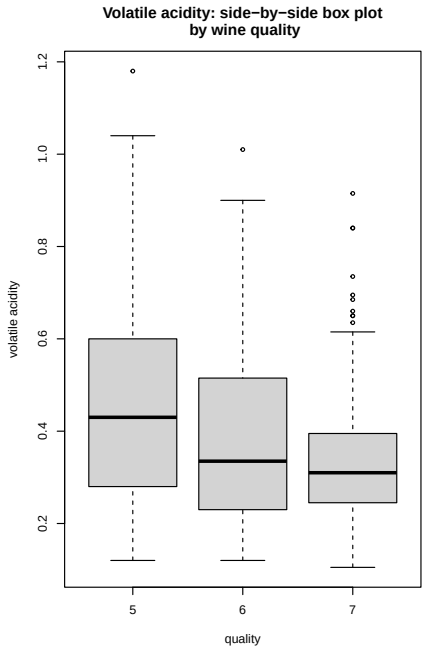
```
## Call:
## lda(Xtraining, grouping = ytraining)
##
## Prior probabilities of groups:
##         0         1
## 0.5033333 0.4966667
##
## Group means:
##   fixed.acidity volatile.acidity citric.acid residual.sugar  chlorides
## 0    -0.5234793       -0.5625877   0.1178026      0.4623053 -0.5670536
## 1     0.4554586        0.6513935  -0.2080615     -0.4703329  0.6280753
##   free.sulfur.dioxide total.sulfur.dioxide    density         pH  sulphates
## 0           0.5114840            0.7680373 -0.5148798 -0.4528581 -0.5299193
## 1          -0.5605023           -0.7483006  0.4666425  0.4127351  0.5224242
##       alcohol
## 0  0.01078592
## 1 -0.01771826
##
## Coefficients of linear discriminants:
##                              LD1
## fixed.acidity        -0.78459748
## volatile.acidity      0.28730033
## citric.acid          -0.17449733
## residual.sugar       -1.59412246
## chlorides             0.17489943
## free.sulfur.dioxide   0.17595964
## total.sulfur.dioxide -0.94957375
## density               2.74796601
## pH                   -0.12413857
## sulphates             0.09451644
## alcohol               1.09904526

## [1] 0.003333333
```

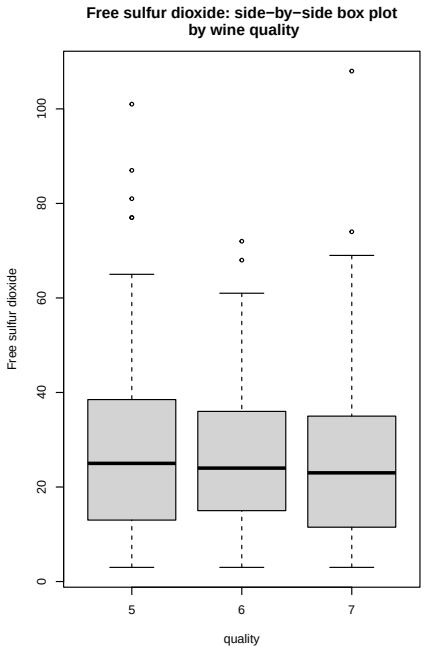
Figure 3.3: Using K-means clustering to predict wine type

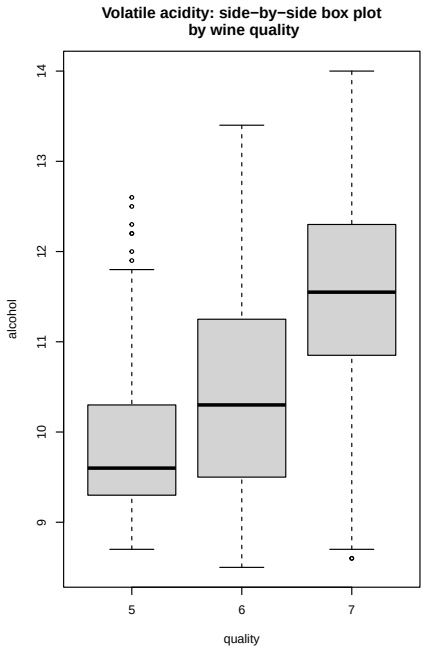



```
## [1] 0.01666667
```

Figure 4.1.1: Side by side boxplot of volatile acidity, free sulfur dioxide, and alcohol by quality





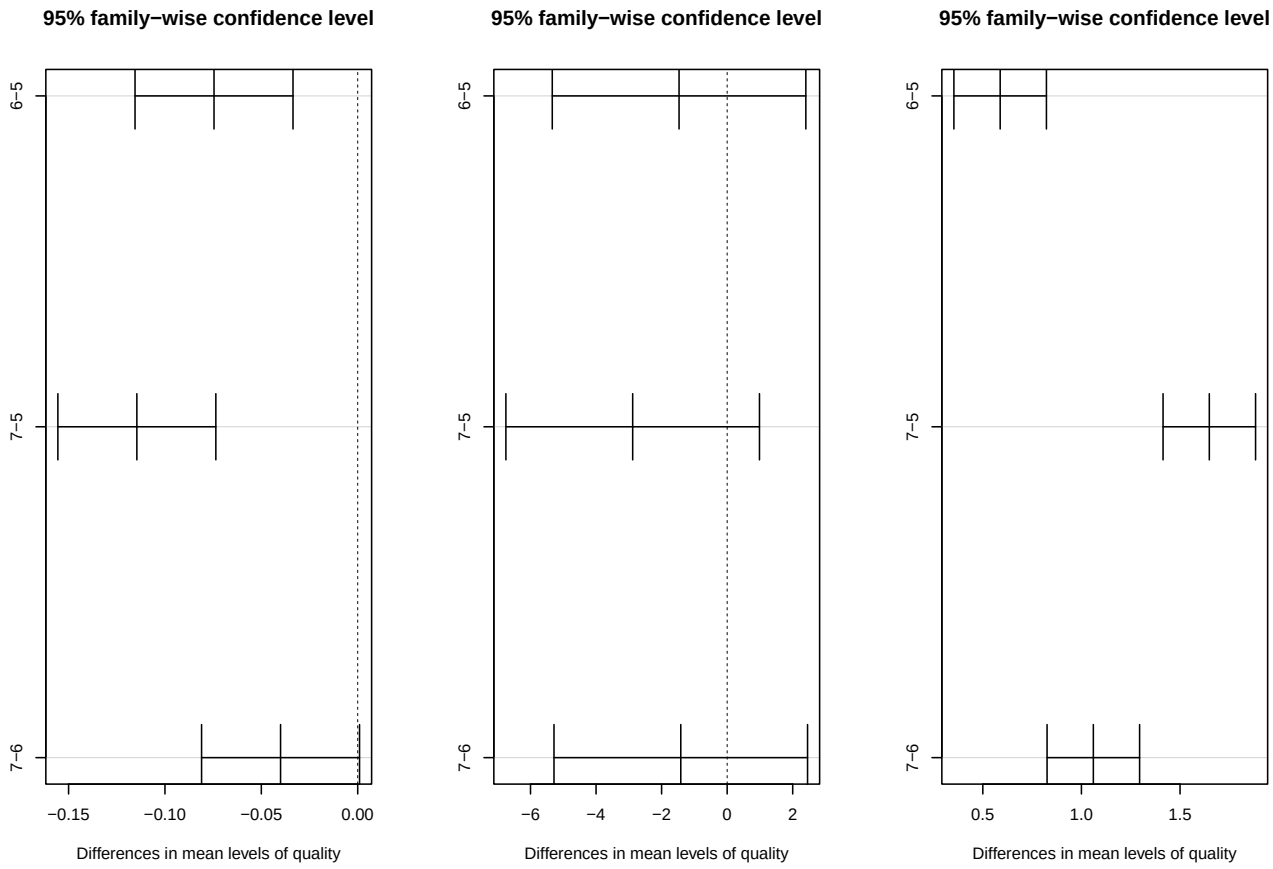


Fit Model

Table 4.1.1: Result of Tukey HSD.

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = volatile.acidity ~ quality, data = wine)
##
## $quality
##          diff         lwr           upr     p adj
## 6-5 -0.074550 -0.11552633 -0.0335736658 0.0000661
## 7-5 -0.114575 -0.15555133 -0.0735986658 0.0000000
## 7-6 -0.040025 -0.08100133  0.0009513342 0.0572570

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = free.sulfur.dioxide ~ quality, data = wine)
##
## $quality
##        diff       lwr       upr     p adj
## 6-5 -1.4675 -5.335887 2.4008867 0.6459968
## 7-5 -2.8825 -6.750887 0.9858867 0.1873612
## 7-6 -1.4150 -5.283387 2.4533867 0.6660948

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = alcohol ~ quality, data = wine)
##
## $quality
##         diff       lwr       upr p adj
## 6-5 0.588000 0.3534593 0.8225407     0
## 7-5 1.648383 1.4138426 1.8829240     0
## 7-6 1.060383 0.8258426 1.2949240     0
```

Figure 4.1.2: Tukey HSD plot of volatile acidity, free sulfur dioxide, and alcohol



Fit Model

Table 4.1.2: Summary of Multinomial Logistic Regression Model.

```
## # weights:  15 (8 variable)
## initial  value 527.333899
## iter  10 value 429.770753
## final  value 429.661504
## converged
```

```
## Call:
## nnet::multinom(formula = quality ~ volatile.acidity + free.sulfur.dioxide +
##     alcohol, data = w_train)
##
## Coefficients:
##   (Intercept) volatile.acidity free.sulfur.dioxide   alcohol
## 6   -4.863244        -2.617994         -0.01350241 0.6169746
## 7  -13.181848        -4.195441         -0.01621638 1.4308503
##
## Std. Errors:
##   (Intercept) volatile.acidity free.sulfur.dioxide   alcohol
## 6    1.407597        0.6859287         0.007544570 0.1300807
## 7    1.623931        0.8525721         0.009028515 0.1464531
##
## Residual Deviance: 859.323
## AIC: 875.323
```

Table 4.1.3: Summary of predictions and model accuracy.

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:MASS':
##
##     select

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## [1] 5 5 5 7 5 5
## Levels: 5 6 7

## [1] 0.6583333
```

Table 4.2: Result of LDA in distinguishing high quality of wines and other.

```
## Call:
## lda(Xtraining, grouping = ytraining)
##
## Prior probabilities of groups:
##         0         1
## 0.6633333 0.3366667
##
## Group means:
##   fixed.acidity volatile.acidity citric.acid residual.sugar  chlorides
## 0   -0.09332592        0.1834869  -0.1784270      0.0801465  0.1281017
## 1    0.07316646       -0.2416534   0.2207327     -0.1606006 -0.1736051
##   free.sulfur.dioxide total.sulfur.dioxide    density          pH   sulphates
## 0         -0.01318271            0.0992217  0.1839921 -0.04104690 -0.09284254
## 1         -0.03621200           -0.1511710 -0.4438766  0.01271578  0.16137688
##      alcohol
## 0 -0.3999660
## 1  0.7780385
##
## Coefficients of linear discriminants:
##                              LD1
## fixed.acidity         0.88427416
## volatile.acidity     -0.16708982
## citric.acid           0.08065936
## residual.sugar        0.70425327
## chlorides             0.03101315
## free.sulfur.dioxide   0.36409925
## total.sulfur.dioxide -0.42869401
## density              -0.96224219
## pH                    0.41978467
## sulphates             0.20216003
## alcohol               0.66800444

## [1] 0.2166667
```

Appendix: R Script

```
## For Table 1:
wine = read.csv("wine.csv")
stat_analy = round(cbind(sapply(wine, mean),sapply(wine, median),sapply(wine, sd)),3)
stat_analy = cbind(sapply(wine, typeof), stat_analy)
colnames(stat_analy) = c('Type','Mean','Median','SD')
knitr::kable(stat_analy)
par(mfrow=c(2,2))
hist(wine$fixed.acidity, main="Histogram for fixed.acidity", xlab="fixed.acidity")
hist(wine$volatile.acidity, main="Histogram for volatile.acidity", xlab="volatile.acidity")
hist(wine$citric.acid, main="Histogram for citric.acid", xlab="citric.acid")
hist(wine$residual.sugar, main="Histogram for residual.sugar", xlab="residual.sugar")

par(mfrow=c(3,3))
hist(wine$chlorides, main="Histogram for chlorides", xlab="chlorides")
hist(wine$free.sulfur.dioxide, main="Histogram for free.sulfur.dioxide", xlab="free.sulfur.dioxide")
hist(wine$total.sulfur.dioxide, main="Histogram for total.sulfur.dioxide", xlab="total.sulfur.dioxide")
hist(wine$density, main="Histogram for density", xlab="density")
hist(wine$pH, main="Histogram for pH", xlab="pH")
hist(wine$sulphates, main="Histogram for sulphates", xlab="sulphates")
hist(wine$alcohol, main="Histogram for alcohol", xlab="alcohol")
hist(wine$quality, main="Histogram for quality", xlab="quality")
pie(table(wine$red), labels = c('0','1'), main="Pie chart for RED")
z_scores = scale(wine[, c("chlorides", "sulphates")])
outlier_index = which(rowSums(abs(z_scores) > 3) > 0)
wine1 = wine
wine = wine[-outlier_index, ]

cor_wine = cor(wine)

panel.cor = function(x, y){
  par(usr = c(0, 1, 0, 1))
  r = round(cor(x, y, use="complete.obs"), 2)
  txt = paste0("R = ", r)
  text(0.5, 0.5, txt, cex = 2.5 * r)
}

##pairs(~fixed.acidity+volatile.acidity+citric.acid+residual.sugar+chlorides+free.sulfur.dioxide+total.sulf
##      data=wine,
##      labels = c("fixed acidity","volatile acidity","citric acid","residual sugar","chlorides","free sulf
##                 "total sulfur\ndioxide","density","pH","sulphates","alcohol","quality","red"),
##      lower.panel = panel.cor,
##      cex.labels = 1.2)

cor_x_wine = cor_wine[1:11,1:11]

high_cor_pairs = which((cor_x_wine>= 0.5 & cor_x_wine< 1 & upper.tri(cor_x_wine)), arr.ind = TRUE)
high_cor_pairs_names = t(rbind(colnames(wine[high_cor_pairs[,1]]),colnames(wine[high_cor_pairs[,2]])))
high_cor = cor_wine[high_cor_pairs]

par(mfrow=c(2,2))
for (i in c(1:4)) {
  plot(wine[,high_cor_pairs[i,1]],wine[,high_cor_pairs[i,2]],
     xlab=colnames(wine)[high_cor_pairs[i,1]],
```

```
     ylab=colnames(wine)[high_cor_pairs[i,2]],
     main=paste(colnames(wine)[high_cor_pairs[i,2]], " - ", colnames(wine)[high_cor_pairs[i,1]],
                "\nCor =", round(cor_wine[high_cor_pairs[i,1],high_cor_pairs[i,2]],digits = 3)))
}
pca_result = prcomp(wine[,1:11], scale. = TRUE)

sum((pca_result$sdev^2 / sum(pca_result$sdev^2))[1:2])

PC1 = pca_result$x[, 1]
PC2 = pca_result$x[, 2]

X = cbind(PC1, PC2)
n = dim(X)[1]
p = dim(X)[2]
S = cov(X)
centeredX = scale(X, scale=FALSE)

par(mfrow=c(1,2))
plot(pca_result$sdev^2, type="b", xlab="Dimension",  ylab="Eigenvalue",
     ylim=1.1*c(0, max(pca_result$sdev^2)))
screeplot(pca_result,main='')
par(mfrow = c(1,3))
qqnorm(PC1, main = "Q-Q Plot\n1st Principal Components")
qqline(PC1, col = "red")

qqnorm(PC2, main = "Q-Q Plot\n2nd Principal Components")
qqline(PC1, col = "red")

theoQ = qchisq(((1:n)-0.5)/n, p)
sampQ = sort(diag(centeredX%*%solve(S)%*%t(centeredX)))
plot(theoQ, sampQ,
     xlab="Theoretical Quantiles",
     ylab="Sample Quantiles",
     main=expression(paste(chi^2," Q-Q Plot")))
### quantile-quantile plots for assessing normality
## marginal normal q-q plots
wine <- read.csv("wine.csv")
PCA <- prcomp(wine)
X <- as.data.frame(PCA$x[,1:2])

par(mfrow=c(1,2))
qqnorm(X[,1],main=expression(paste(chi^2, " Q-Q Plot for PC1")))
qqnorm(X[,2],main=expression(paste(chi^2, " Q-Q Plot for PC2")))

## chi squared q-q plot for multivariate data
n <- dim(X)[1]
p <- dim(X)[2]
S <- cov(X)
centeredX <- scale(X, scale=FALSE)

par(mfrow=c(1,1))
theoQ <- qchisq(((1:n)-0.5)/n, p)
sampQ <- sort(diag(centeredX%*%solve(S)%*%t(centeredX)))
plot(theoQ, sampQ,
     xlab="Theoretical Quantiles",
     ylab="Sample Quantiles",
```

```r
     main=expression(paste(chi^2," Q-Q Plot")))
## loading data
data <- read.csv("wine.csv")
head(data)
set.seed(123)
num_rows <- nrow(data)

# one-way MANOVA
Y <- as.matrix(data[,1:11])
red <- data$red

fittedModel <- manova(Y ~ red)
anova(fittedModel, test="Wilks")
#the result shows significant difference between red wine and white wine.

#PCA
PCA <- prcomp(data[,1:11])

## scree plots
plot(PCA$sdev^2,
     type="b",
     xlab="Principal component",
     ylab="Eigenvalue",
     main="Scree plot,\n unscaled data")
#the scree plots shows that we should use first 2 principal component.

#get the first two Principal component
new_variables <- PCA$x[, 1:2]

#plot the Principal components
plot(new_variables)
#use K-means cluster to predict whether it is red wine or white wine
kmeans_result <- kmeans(new_variables, centers = 2)

# get the cluster labels of the K-means cluster
cluster_labels <- kmeans_result$cluster

#Draw scatter plots, coloring according to cluster labels
plot(new_variables, col = cluster_labels, pch = 19, main = "K-means Clustering")

#the real category of wine
new=cbind(new_variables,data[,13])
new=as.data.frame(new)
new$V3=as.factor(new$V3)
plot(new$PC1, new$PC2, col = as.numeric(new$V3), pch = 19,
     xlab = "PC1", ylab = "PC2", main = "real category of wine")
legend("topright", legend = levels(new$V3),
       col = 1:length(levels(new$V3)), pch = 19, title = "red or white")

#calculate the error rate
kmeans_result <- kmeans(new[, c("PC1", "PC2")], centers = 2)
new$Cluster <- factor(kmeans_result$cluster, labels = c(0, 1))
#The number of values predicted incorrectly
different_values_count <- sum(new$V3 != new$Cluster)
n <- nrow(new)
error_rate=different_values_count/n#error_rate=0.0975
```

```r
#This means that after dimensionality reduction of the data by principal component analysis,
#the accuracy of wine category prediction by k-means clustering method is 90.25%.
# LDA to predict red
#### with tran/valid split
library(MASS)
library(klaR)
X <- scale(wine[,1:11])
y <- as.numeric(wine$red==1)
set.seed(498022)
ind <- sample(dim(X)[1], round(dim(X)[1]/2))
Xtraining <- X[ind,]
Xtesting <- X[-ind,]
ytraining <- y[ind]
ytesting <- y[-ind]

ldaFit <- lda(Xtraining, ytraining)
ldaFit

prd <- predict(ldaFit, Xtesting)

tab <- table(Predicted = prd$class, Actual = ytesting)
print(1-sum(diag(tab))/sum(tab)) # misclassification rate
# K-means clustering
# red predict
set.seed(24118)
num_std <- scale(wine[,1:11])
y2 <- as.numeric(wine$red==1)
cls2 <- kmeans(num_std, 2)
par(mfrow=c(1,2))
plot(num_std, col=cls2$cluster, main='cls2$cluster')
plot(num_std, col=y2+1, main='y2+1')
tab <- table(Predicted = cls2$cluster, Actual = y2)
1-sum(diag(tab))/sum(tab) # misclassification rate
# 4.1
dim(wine)
sapply(wine, class)
wine <- wine[,1:12]
wine$quality <- as.factor(wine$quality)
# Generate indices for train and test data
set.seed(23)
n = nrow(wine)
train_index = sample.int(n, size = 0.8*n)
w_train = wine[train_index, ]
w_valid = wine[-train_index, ]
par(mfrow=c(1,3))
boxplot(wine$volatile.acidity~wine$quality,main='Volatile acidity: side-by-side box plot\n by wine quality'
xlab='quality',ylab='volatile acidity')
boxplot(wine$free.sulfur.dioxide~wine$quality,main='Free sulfur dioxide: side-by-side box plot\n by wine qu
boxplot(wine$alcohol~wine$quality,main='Volatile acidity: side-by-side box plot\n by wine quality',
xlab='quality',ylab='alcohol')
library(ggplot2)
T1 <- TukeyHSD(aov(volatile.acidity ~ quality, wine)); T1
T2 <- TukeyHSD(aov(free.sulfur.dioxide ~ quality, wine)) ;T2
T3 <- TukeyHSD(aov(alcohol ~ quality, wine)); T3
par(mfrow=c(1,3))
plot(T1)
```

```r
plot(T2)
plot(T3)
# Fit the model
model <- nnet::multinom(quality ~ volatile.acidity + free.sulfur.dioxide + alcohol, data = w_train)
# Summarize the model
summary(model)
# Make predictions
library(dplyr)
predicted.classes <- model %>% predict(w_valid)
head(predicted.classes)
# Model accuracy
mean(predicted.classes == w_valid$quality)
# LDA to predict wine quality
#### with tran/valid split
library(klaR)
X <- scale(wine[,1:11])
y <- as.numeric(wine$quality==7)
set.seed(498022)
ind <- sample(dim(X)[1], round(dim(X)[1]/2))
Xtraining <- X[ind,]
Xtesting <- X[-ind,]
ytraining <- y[ind]
ytesting <- y[-ind]

ldaFit <- lda(Xtraining, ytraining)
ldaFit

prd <- predict(ldaFit, Xtesting)

tab <- table(Predicted = prd$class, Actual = ytesting)
print(1-sum(diag(tab))/sum(tab)) # misclassification rate
```