STAT350 Final Project - Team 16

# Abstract

This exploratory data analysis will be examining data pertaining to the red variety of the Portugese “Vinho Verde” wine. Our analysis will begin by graphically representing the data in order to learn more about it. By constructing visualizations, we will determine the types of relationships between the independent variables and the dependent variable quality. From there, we will be fitting the data to a linear model, and analyzing the residuals. We will then perform variable selection to end up with a model that contains only significant independent variables. Finally, we will perform cross validation to determine the efficacy of our model. By constructing a model containing significant variables, we hope to determine which physiochemical variables are most important in determining the quality of a wine.

# Introduction

The “Vinho Verde” red wine dataset contains 12 variables and 1599 unique observations. Due to privacy limitations, this data only contains the output variable quality and 11 physiochemical variables. All of the physiochemical variables are of the numeric class, while the dependent variable quality is an integer.

About the Variables

* Fixed acidity: Fixed or nonvolatile acids involved with wine
* Volatile acidity: Amount of acetic acid in wine - large amounts may give wine a vinegary taste
* Citric acid: Acid that adds freshness and flavor to wines, normally found in small quantities
* Residual sugar: Amount of sugar remaining after fermentation stops
* Chlorides: Amount of salt in the wine
* Free sulfur dioxide: Free form of sulfur dioxide that prevents microbial growth and the oxidation of wine
* Total sulfur dioxide: Amount of free and bound forms of sulfur dioxide
* Density: Density of water depending on the percent alcohol and sugar content
* pH: Acidity of a wine on a scale from 0 to 14 - most wines are between 3 or 4
* Sulphates: Wine additive that acts as an antimicrobial and antioxidant
* Alcohol: Percent alcohol content of wine

Libraries

Here are the libraries that we used in this project.

library(latexpdf)  
library(stringr)  
library(ggplot2)  
library(gridExtra)  
library(faraway)  
library(MASS)  
library(olsrr)  
library(Metrics)  
library(caret)

# Data Description

About the Data

First, we will read the red wine data and look at the fundamentals of the data.

red.wine = read.csv("winequality-red.csv", header=TRUE, sep=";")  
head(red.wine)

## fixed.acidity volatile.acidity citric.acid residual.sugar chlorides  
## 1 7.4 0.70 0.00 1.9 0.076  
## 2 7.8 0.88 0.00 2.6 0.098  
## 3 7.8 0.76 0.04 2.3 0.092  
## 4 11.2 0.28 0.56 1.9 0.075  
## 5 7.4 0.70 0.00 1.9 0.076  
## 6 7.4 0.66 0.00 1.8 0.075  
## free.sulfur.dioxide total.sulfur.dioxide density pH sulphates alcohol  
## 1 11 34 0.9978 3.51 0.56 9.4  
## 2 25 67 0.9968 3.20 0.68 9.8  
## 3 15 54 0.9970 3.26 0.65 9.8  
## 4 17 60 0.9980 3.16 0.58 9.8  
## 5 11 34 0.9978 3.51 0.56 9.4  
## 6 13 40 0.9978 3.51 0.56 9.4  
## quality  
## 1 5  
## 2 5  
## 3 5  
## 4 6  
## 5 5  
## 6 5

names(red.wine)

## [1] "fixed.acidity" "volatile.acidity" "citric.acid"   
## [4] "residual.sugar" "chlorides" "free.sulfur.dioxide"   
## [7] "total.sulfur.dioxide" "density" "pH"   
## [10] "sulphates" "alcohol" "quality"

dim(red.wine)

## [1] 1599 12

sapply(red.wine, class)

## fixed.acidity volatile.acidity citric.acid   
## "numeric" "numeric" "numeric"   
## residual.sugar chlorides free.sulfur.dioxide   
## "numeric" "numeric" "numeric"   
## total.sulfur.dioxide density pH   
## "numeric" "numeric" "numeric"   
## sulphates alcohol quality   
## "numeric" "numeric" "integer"

red.info = summary(red.wine)  
red.info

## fixed.acidity volatile.acidity citric.acid residual.sugar   
## Min. : 4.60 Min. :0.1200 Min. :0.000 Min. : 0.900   
## 1st Qu.: 7.10 1st Qu.:0.3900 1st Qu.:0.090 1st Qu.: 1.900   
## Median : 7.90 Median :0.5200 Median :0.260 Median : 2.200   
## Mean : 8.32 Mean :0.5278 Mean :0.271 Mean : 2.539   
## 3rd Qu.: 9.20 3rd Qu.:0.6400 3rd Qu.:0.420 3rd Qu.: 2.600   
## Max. :15.90 Max. :1.5800 Max. :1.000 Max. :15.500   
## chlorides free.sulfur.dioxide total.sulfur.dioxide density   
## Min. :0.01200 Min. : 1.00 Min. : 6.00 Min. :0.9901   
## 1st Qu.:0.07000 1st Qu.: 7.00 1st Qu.: 22.00 1st Qu.:0.9956   
## Median :0.07900 Median :14.00 Median : 38.00 Median :0.9968   
## Mean :0.08747 Mean :15.87 Mean : 46.47 Mean :0.9967   
## 3rd Qu.:0.09000 3rd Qu.:21.00 3rd Qu.: 62.00 3rd Qu.:0.9978   
## Max. :0.61100 Max. :72.00 Max. :289.00 Max. :1.0037   
## pH sulphates alcohol quality   
## Min. :2.740 Min. :0.3300 Min. : 8.40 Min. :3.000   
## 1st Qu.:3.210 1st Qu.:0.5500 1st Qu.: 9.50 1st Qu.:5.000   
## Median :3.310 Median :0.6200 Median :10.20 Median :6.000   
## Mean :3.311 Mean :0.6581 Mean :10.42 Mean :5.636   
## 3rd Qu.:3.400 3rd Qu.:0.7300 3rd Qu.:11.10 3rd Qu.:6.000   
## Max. :4.010 Max. :2.0000 Max. :14.90 Max. :8.000

table(is.na(red.wine))

##   
## FALSE   
## 19188

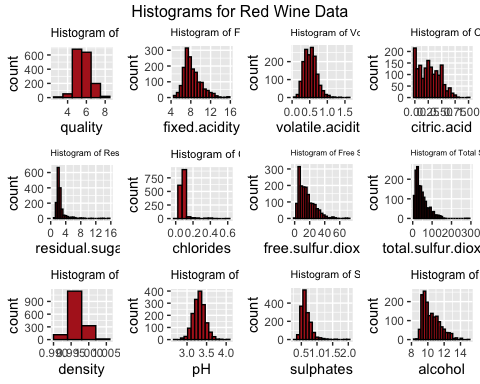
Include some basic descriptions including the class and the descriptive statistics of the variables. + Missing value

Data Visualization - Histograms

Data visualization is one of the simplest way to understand our dataset. Histograms or boxplots plot not only the shape of individual variables but also the relationship between them. After this process, we will be able to keep in mind the approximate distribution of each regressors.

First, we will generate histograms for the all the variables. This will tell us the approximate distribution of each variables.

hist\_y = ggplot(aes(quality), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 1) +  
 ggtitle('Histogram of Quality') + theme(plot.title = element\_text(size=9))  
  
hist\_x1 = ggplot(aes(fixed.acidity), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 0.6) +  
 ggtitle('Histogram of Fixed Acidity') + theme(plot.title = element\_text(size=8))  
  
hist\_x2 = ggplot(aes(volatile.acidity), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 0.08) +  
 ggtitle('Histogram of Volatile Acidity') + theme(plot.title = element\_text(size=7.5))  
  
hist\_x3 = ggplot(aes(citric.acid), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 0.05) +  
 ggtitle('Histogram of Citric Acid') + theme(plot.title = element\_text(size=8))  
  
hist\_x4 = ggplot(aes(residual.sugar), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 0.5) +  
 ggtitle('Histogram of Residual Sugar') + theme(plot.title = element\_text(size=6.7))  
  
hist\_x5 = ggplot(aes(chlorides), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 0.05) +  
 ggtitle('Histogram of Chlorides') + theme(plot.title = element\_text(size=8.5))  
  
hist\_x6 = ggplot(aes(free.sulfur.dioxide), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 3) +  
 ggtitle('Histogram of Free Sulfur Dioxide') + theme(plot.title = element\_text(size=6))  
  
hist\_x7 = ggplot(aes(total.sulfur.dioxide), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 8.3) +  
 ggtitle('Histogram of Total Sulfur Dioxide') + theme(plot.title = element\_text(size=6))  
  
hist\_x8 = ggplot(aes(density), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 0.004) +  
 ggtitle('Histogram of Density') + theme(plot.title = element\_text(size=9))  
  
hist\_x9 = ggplot(aes(pH), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 0.09) +  
 ggtitle('Histogram of pH') + theme(plot.title = element\_text(size=9))  
  
hist\_x10 = ggplot(aes(sulphates), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 0.1) +  
 ggtitle('Histogram of Sulphates') + theme(plot.title = element\_text(size=8))  
  
hist\_x11 = ggplot(aes(alcohol), data = red.wine) +  
 geom\_histogram(aes(color=I('black'), fill=I('firebrick')), binwidth = 0.3) +  
 ggtitle('Histogram of Alcohol') + theme(plot.title = element\_text(size=9))  
  
grid.arrange(hist\_y, hist\_x1, hist\_x2, hist\_x3, hist\_x4, hist\_x5, hist\_x6, hist\_x7, hist\_x8, hist\_x9, hist\_x10, hist\_x11, ncol = 4, top="Histograms for Red Wine Data")



There are a couple of things that we can see from the histograms above.

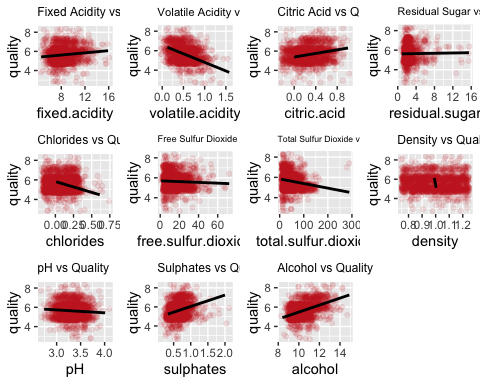
* Quality variable seem to have a normal distribution, with most of the values concentrated in categories 5, 6, and 7. In other words, there are much more normal wines than excellent or poor ones.
* Most of the variables have a very heavy right-tail, except for quality, pH, sulphates, and alcohol.
* Some of the skewed data such as fixed acidity, volatile acidity, or citric acid can be normal when the outliers are detected and fixed.

Data Visualization - Regressors vs. Predictor

Now, we will plot each of the regressors against the predictor. This will help us see the approximate relationship between each varibles and wine quality.

group\_x1 = ggplot(aes(fixed.acidity, quality), data = red.wine) +   
 ggtitle("Fixed Acidity vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=9))  
  
group\_x2 = ggplot(aes(volatile.acidity, quality), data = red.wine) +   
 ggtitle("Volatile Acidity vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=8.5))  
  
group\_x3 = ggplot(aes(citric.acid, quality), data = red.wine) +   
 ggtitle("Citric Acid vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=9))  
  
group\_x4 = ggplot(aes(residual.sugar, quality), data = red.wine) +   
 ggtitle("Residual Sugar vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=8))  
  
group\_x5 = ggplot(aes(chlorides, quality), data = red.wine) +   
 ggtitle("Chlorides vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=9))  
  
group\_x6 = ggplot(aes(free.sulfur.dioxide, quality), data = red.wine) +   
 ggtitle("Free Sulfur Dioxide vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=7))  
  
group\_x7 = ggplot(aes(total.sulfur.dioxide, quality), data = red.wine) +   
 ggtitle("Total Sulfur Dioxide vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=6.5))  
  
group\_x8 = ggplot(aes(density, quality), data = red.wine) +   
 ggtitle("Density vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=9))  
  
group\_x9 = ggplot(aes(pH, quality), data = red.wine) +   
 ggtitle("pH vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=9))  
  
group\_x10 = ggplot(aes(sulphates, quality), data = red.wine) +   
 ggtitle("Sulphates vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=9))  
  
group\_x11 = ggplot(aes(alcohol, quality), data = red.wine) +   
 ggtitle("Alcohol vs Quality") +  
 geom\_jitter(width = 0.25, alpha = 0.1, colour = "firebrick3") +   
 geom\_smooth(method = "lm", se = FALSE, colour = "black") + theme(plot.title = element\_text(size=9))

grid.arrange(group\_x1, group\_x2, group\_x3, group\_x4, group\_x5, group\_x6, group\_x7, group\_x8, group\_x9, group\_x10, group\_x11, ncol = 4)



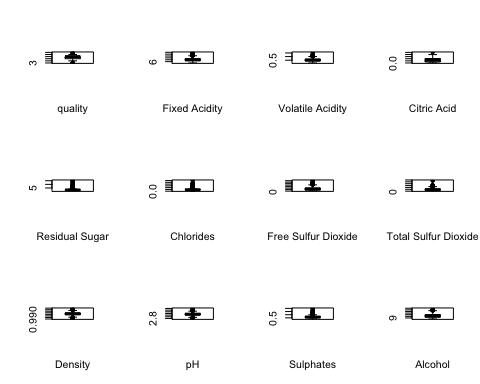
There are a couple of things that we can see from the plots above.

* Fixed acidity, citric acid, sulphates, and alcohol have a positive relationship with quality.
* Volatile acidity, total sulfur dioxide, and pH have a negative relationship with quality.
* Rest of the data shows a fairly random pattern.

Data Visualization - Boxplots

Now, we will generate boxplots of each variable and see more closely of the distribution of variables.

par(mfrow = c(3,4))  
boxplot(red.wine$quality, col="slategray2", pch=19, xlab = "quality")  
boxplot(red.wine$fixed.acidity, col="slategray2", pch=19, xlab = "Fixed Acidity")  
boxplot(red.wine$volatile.acidity, col="slategray2", pch=19, xlab = "Volatile Acidity")  
boxplot(red.wine$citric.acid, col="slategray2", pch=19, xlab = "Citric Acid")  
boxplot(red.wine$residual.sugar, col="slategray2", pch=19, xlab = "Residual Sugar")  
boxplot(red.wine$chlorides, col="slategray2", pch=19, xlab = "Chlorides")  
boxplot(red.wine$free.sulfur.dioxide, col="slategray2", pch=19, xlab = "Free Sulfur Dioxide")  
boxplot(red.wine$total.sulfur.dioxide, col="slategray2", pch=19, xlab = "Total Sulfur Dioxide")  
boxplot(red.wine$density, col="slategray2", pch=19, xlab = "Density")  
boxplot(red.wine$pH, col="slategray2", pch=19, xlab = "pH")  
boxplot(red.wine$sulphates, col="slategray2", pch=19, xlab = "Sulphates")  
boxplot(red.wine$alcohol, col="slategray2", pch=19, xlab = "Alcohol")



There are a couple of things that we can see from the boxplots above.

* Citric acid, free sulfur dioxide, total sulfur dioxide, and alcohol are right-skewed. The skewness of the data would not be fixed even after eliminating the outliers.
* For other variables, detecting the outliers and fixing them will make the distribution more symmetric. However, some of the outliers seem very far away from the data.
* Except for density, most of the outliers are on the larger side of the data.

New Additional Data Point

Before we begin the data analysis, we will introduce the median point as a new additional data point to the red wine data.

red.datapoints = vector(mode = 'numeric', length = 12)  
red.datapoints = c(2.60, 0, 0, 0.3, 0.001, 0, 3, 0.75, 0.740, 0.11, 6.40, 1)  
red.wine = rbind(red.wine, red.datapoints)  
summary(red.wine)

## fixed.acidity volatile.acidity citric.acid residual.sugar   
## Min. : 2.600 Min. :0.0000 Min. :0.0000 Min. : 0.300   
## 1st Qu.: 7.100 1st Qu.:0.3900 1st Qu.:0.0900 1st Qu.: 1.900   
## Median : 7.900 Median :0.5200 Median :0.2600 Median : 2.200   
## Mean : 8.316 Mean :0.5275 Mean :0.2708 Mean : 2.537   
## 3rd Qu.: 9.200 3rd Qu.:0.6400 3rd Qu.:0.4200 3rd Qu.: 2.600   
## Max. :15.900 Max. :1.5800 Max. :1.0000 Max. :15.500   
## chlorides free.sulfur.dioxide total.sulfur.dioxide density   
## Min. :0.00100 Min. : 0.00 Min. : 3.00 Min. :0.7500   
## 1st Qu.:0.07000 1st Qu.: 7.00 1st Qu.: 22.00 1st Qu.:0.9956   
## Median :0.07900 Median :14.00 Median : 38.00 Median :0.9968   
## Mean :0.08741 Mean :15.87 Mean : 46.44 Mean :0.9966   
## 3rd Qu.:0.09000 3rd Qu.:21.00 3rd Qu.: 62.00 3rd Qu.:0.9978   
## Max. :0.61100 Max. :72.00 Max. :289.00 Max. :1.0037   
## pH sulphates alcohol quality   
## Min. :0.74 Min. :0.1100 Min. : 6.40 Min. :1.000   
## 1st Qu.:3.21 1st Qu.:0.5500 1st Qu.: 9.50 1st Qu.:5.000   
## Median :3.31 Median :0.6200 Median :10.20 Median :6.000   
## Mean :3.31 Mean :0.6578 Mean :10.42 Mean :5.633   
## 3rd Qu.:3.40 3rd Qu.:0.7300 3rd Qu.:11.10 3rd Qu.:6.000   
## Max. :4.01 Max. :2.0000 Max. :14.90 Max. :8.000

We now introduced the median point as a new point to the red wine quality data.

The plot above tells us a few things.

There is a high positive correlation between :

* fixed acidity and volatile acidity
* fixed acidity and density
* free sulfur dioxide and total sulfur dioxide

There is a high negative correlation between (from strongest to weakest):

* fixed acidity and pH
* volatile acidity and citric acid
* citric acid and density
* density and alcohol

Keeping in mind that there exists correlation between several variables, let’s begin our multiple linear regression.

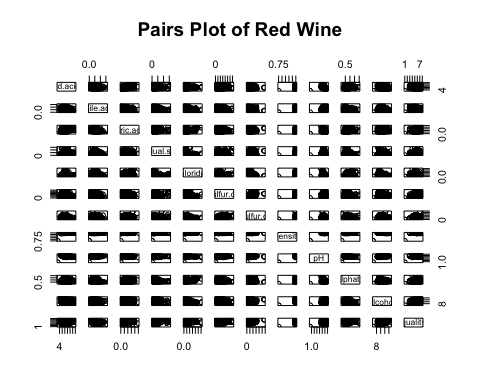
# Methods

## Multiple Linear Regression - Full Model

Pairs Plot

First, with red wine, conduct a pairs plot to see if there are any particular patterns.

pairs(red.wine, main = "Pairs Plot of Red Wine")



The pairs plot tells us a bit about the relationship between variables in the dataset. Specifically, we can see that a linear model seems appropriate, although some of the variables seem to have less of a linear relationship (we can look into that by conducting hypothesis tests). Also, we have to keep in mind of the multicollinearity present among the explanatory variables. Now, we will conduct a linear regression against the quality of the red wine.

Fitting Into a Linear Model

red.mdl = lm(quality~., data=red.wine)  
red.sum = summary(red.mdl)

There are a couple of things that we notice from the summary of the linear model.

* Fixed acidity, residual sugar, free sulfur dioxide, sulphates, and alcohol have a positive relationship with red wine quality.
* Volatile acidity, citric acid, chlorides, total sulfur dioxide, density, and pH have a negative relationship with red wine quality.
* The P-value of the overall regression is significantly small, which suggests that there is a linear association between at least one of the regressors and the red wine quality.
* The adjusted R-squared value is 0.3561 (35.61%), which implies that our linear model fit to the data is not satisfactory. In fact, only 35% of the data can be explained by the model.
* Looking at the individual P-values, fixed acidity, citric acid, residual sugar and density seem to be insignificant given that all the other variables are in the model.

Anova Test

In order to see if some of the regressors are insignificant to the regression, we will first run the anova test.

anova(red.mdl)

## Analysis of Variance Table  
##   
## Response: quality  
## Df Sum Sq Mean Sq F value Pr(>F)   
## fixed.acidity 1 19.10 19.103 45.4264 2.211e-11 \*\*\*  
## volatile.acidity 1 132.44 132.438 314.9289 < 2.2e-16 \*\*\*  
## citric.acid 1 0.00 0.000 0.0000 1.0000   
## residual.sugar 1 0.28 0.276 0.6575 0.4176   
## chlorides 1 12.20 12.199 29.0079 8.290e-08 \*\*\*  
## free.sulfur.dioxide 1 2.08 2.082 4.9518 0.0262 \*   
## total.sulfur.dioxide 1 30.53 30.530 72.5992 < 2.2e-16 \*\*\*  
## density 1 15.02 15.015 35.7058 2.828e-09 \*\*\*  
## pH 1 1.02 1.017 2.4179 0.1202   
## sulphates 1 50.84 50.840 120.8939 < 2.2e-16 \*\*\*  
## alcohol 1 132.34 132.340 314.6957 < 2.2e-16 \*\*\*  
## Residuals 1588 667.80 0.421   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

The anova test conducts a partial F-test, and supports our hypothesis that citric acid and residual sugar variable seem to be insiginificant to the linear model given that the variables before them are already included in the model.

To check which of the variables must be selected in building the model, we will conduct a variable selection later on.

Multicollinearity

In the pairs plot above, we suspected that there may be near linear relationship between some of the regressor variables in the data. We will conduct a diagnostics since multicollinearity will cause a serious problem that may dramatically impact the usefulness of the linear model. We will use the variance inflation factor (VIF) to check.

vif(red.mdl)

## fixed.acidity volatile.acidity citric.acid   
## 3.533799 1.771331 3.131646   
## residual.sugar chlorides free.sulfur.dioxide   
## 1.114620 1.477776 1.953728   
## total.sulfur.dioxide density pH   
## 2.181671 1.857333 2.958244   
## sulphates alcohol   
## 1.356765 1.349059

Although VIF for fixed acidity are large, but it is not larger than 10, the rule-of-thumb that our textbook suggests. In fact, none of the VIFs are larger than 10, so we will conclude that there are no serious issues regarding multicollinearity.

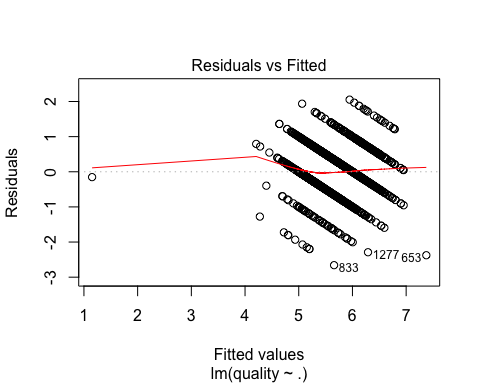
## Model Assessment

Residual Analysis

We will first go over the graphic analysis of the residuals to check if the error are i.i.d. normally distributed, with zero mean and constant variance.

##### Residual vs. Fitted

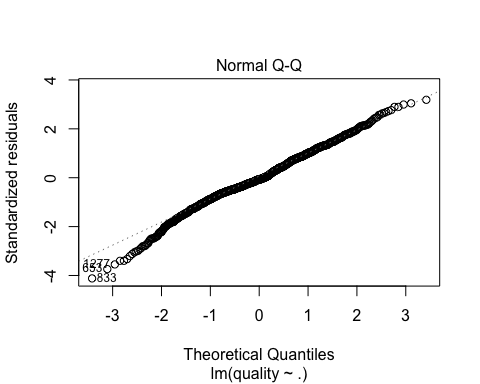
plot(red.mdl, which=1)



Comments on the residual vs. fitted plot

##### Normal Q-Q

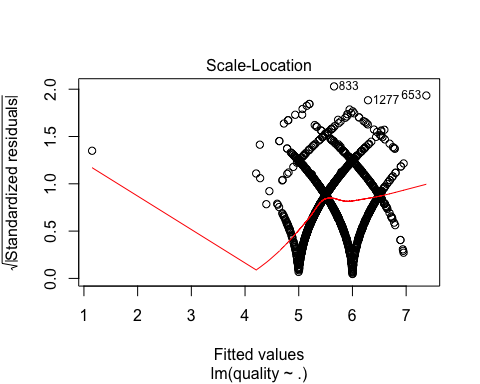
plot(red.mdl, which=2)



We observe that the residuals for our model meet the normality assumption. There seems to be a slight negative skew to the distribution of the residuals, although it is not too much of a concern. Also, there are some potential influential points. We will look at this in a moment.

##### Scale-Location

plot(red.mdl, which=3)

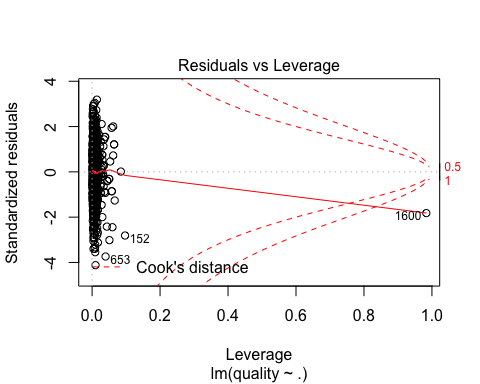


Comments on the Scale-Location Plot

##### Residuals vs. Leverage

plot(red.mdl, which=5)

## Warning in sqrt(crit \* p \* (1 - hh)/hh): NaNs produced  
  
## Warning in sqrt(crit \* p \* (1 - hh)/hh): NaNs produced

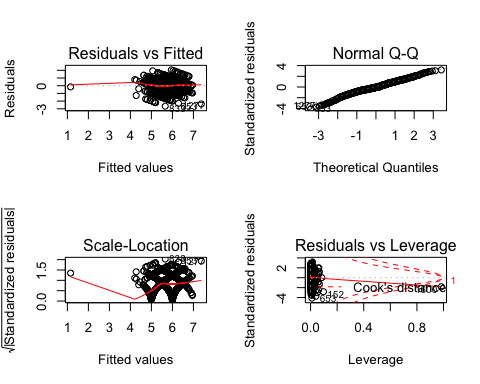


Comments on the Residuals vs. Leverage Plot

##### Overall Residual Plots

par(mfrow=c(2,2))  
plot(red.mdl, which=1)  
plot(red.mdl, which=2)  
plot(red.mdl, which=3)  
plot(red.mdl, which=5)

## Warning in sqrt(crit \* p \* (1 - hh)/hh): NaNs produced  
  
## Warning in sqrt(crit \* p \* (1 - hh)/hh): NaNs produced



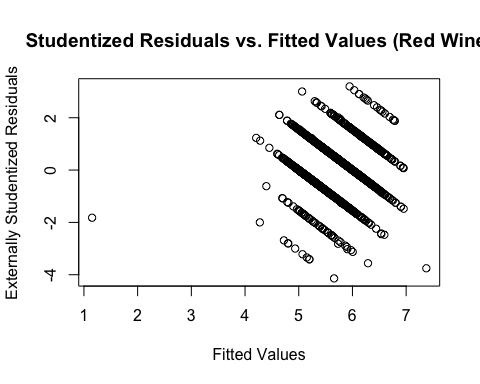
Comments on the Residuals vs. Leverage Plot

Outlier Detection

Outliers are data points that are not typical of the rest of the data. They can either improve or worsen the fit of the equation, so it is important to investigate each outliers or potential influential points.

For the outlier detection, we are going to use externally studentized residuals and plot them against various values.

ti = rstudent(red.mdl)  
plot(red.mdl$fitted.values, ti, xlab = "Fitted Values", ylab="Externally Studentized Residuals",  
 main="Studentized Residuals vs. Fitted Values (Red Wine)")



Identifying the points on the plot, 1600th and 833rd observations potential outliers. To investigate the influence of these points on the model, we will obtain an equation with these two observations deleted.

red.wine.out = red.wine[-c(1600,833), ]  
red.mdl.out = lm(quality ~ . , data=red.wine.out)  
red.sum = summary(red.mdl)  
red.sum.out = summary(red.mdl.out)  
red.mse = mean(red.sum$residuals^2)  
red.mse.out = mean(red.sum.out$residuals^2)  
red.sum

##   
## Call:  
## lm(formula = quality ~ ., data = red.wine)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.65794 -0.36368 -0.04403 0.45568 2.05281   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.624e+01 3.049e+00 -5.328 1.14e-07 \*\*\*  
## fixed.acidity -1.001e-02 1.746e-02 -0.573 0.56658   
## volatile.acidity -1.111e+00 1.203e-01 -9.239 < 2e-16 \*\*\*  
## citric.acid -1.845e-01 1.473e-01 -1.253 0.21053   
## residual.sugar 2.469e-04 1.214e-02 0.020 0.98377   
## chlorides -1.928e+00 4.186e-01 -4.606 4.44e-06 \*\*\*  
## free.sulfur.dioxide 4.672e-03 2.166e-03 2.157 0.03117 \*   
## total.sulfur.dioxide -3.342e-03 7.280e-04 -4.591 4.75e-06 \*\*\*  
## density 2.103e+01 3.426e+00 6.139 1.05e-09 \*\*\*  
## pH -5.857e-01 1.668e-01 -3.510 0.00046 \*\*\*  
## sulphates 8.666e-01 1.111e-01 7.799 1.12e-14 \*\*\*  
## alcohol 3.123e-01 1.760e-02 17.740 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.6485 on 1588 degrees of freedom  
## Multiple R-squared: 0.3722, Adjusted R-squared: 0.3678   
## F-statistic: 85.57 on 11 and 1588 DF, p-value: < 2.2e-16

red.sum.out

##   
## Call:  
## lm(formula = quality ~ ., data = red.wine.out)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.48879 -0.36600 -0.05228 0.44980 2.03022   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.466e+01 2.109e+01 1.169 0.2425   
## fixed.acidity 3.101e-02 2.585e-02 1.199 0.2306   
## volatile.acidity -1.084e+00 1.205e-01 -9.000 < 2e-16 \*\*\*  
## citric.acid -1.822e-01 1.464e-01 -1.244 0.2136   
## residual.sugar 1.606e-02 1.492e-02 1.076 0.2822   
## chlorides -1.815e+00 4.173e-01 -4.348 1.46e-05 \*\*\*  
## free.sulfur.dioxide 4.847e-03 2.163e-03 2.241 0.0252 \*   
## total.sulfur.dioxide -3.331e-03 7.251e-04 -4.594 4.70e-06 \*\*\*  
## density -2.077e+01 2.153e+01 -0.965 0.3348   
## pH -3.653e-01 1.910e-01 -1.913 0.0559 .   
## sulphates 9.252e-01 1.138e-01 8.133 8.38e-16 \*\*\*  
## alcohol 2.724e-01 2.636e-02 10.332 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.6447 on 1586 degrees of freedom  
## Multiple R-squared: 0.3633, Adjusted R-squared: 0.3589   
## F-statistic: 82.28 on 11 and 1586 DF, p-value: < 2.2e-16

red.mse

## [1] 0.4173777

red.mse.out

## [1] 0.4124534

Deleting the potential outlier points has almost no effect on the estimates of the regression coefficients nor on the residual mean square. In fact, deleting the points causes a slight increase in , even though the increase does not seem significant. Hence, we conclude that points 653 and 833 are not influential.

Based on the information we acquired from both the residuals vs. fitted values, we may say that the deficiency of the model is due to trying to fit a discrete quality value to a continuous data.

Outlier Diagnostics : Cook’s Distance

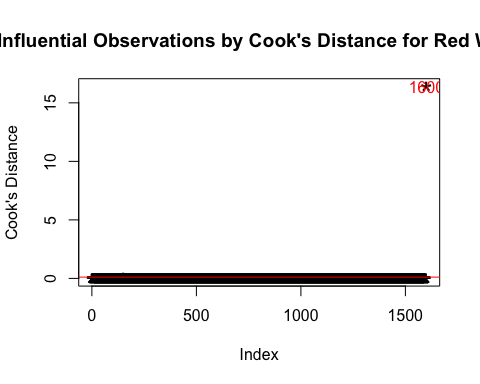
Cook’s distance is one of the ways to measure a point’s influence by considering both the location of a point in the x space and the response variable.

Points with large values of can be interpreted as an influential point.

red.cooksd = cooks.distance(red.mdl)  
max.di = max(red.cooksd)  
max.di

## [1] 16.40488

plot(red.cooksd, pch="\*", cex=2, ylab = "Cook's Distance", main="Influential Observations by Cook's Distance for Red Wine")   
abline(h = 10\*mean(red.cooksd, na.rm=T), col="red") # adding the cutoff line  
text(x=1:length(red.cooksd)+1, y=red.cooksd, labels=ifelse(red.cooksd>10\*mean(red.cooksd, na.rm=T),names(red.cooksd),""), col="red")



When interpreting Cook’s distance, we categorize the values as follows:

* > 0.5 : The data point is worthy of further investigation as influential
* > 1 : The data point is quite likely to be influential
* If sticks out from the other values, it is most likely to be influential

Even though the maximum value for the Cook’s distance is 0.06885, which is even smaller than 0.5, but since it is still larger than other values, we will take a closer look.

red.wine.inf = red.wine[-1600, ]  
red.mdl.inf = lm(quality ~ . , data=red.wine.inf)  
red.sum.inf = summary(red.mdl.inf)  
red.mse.inf = mean(red.sum.inf$residuals^2)  
red.sum

##   
## Call:  
## lm(formula = quality ~ ., data = red.wine)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.65794 -0.36368 -0.04403 0.45568 2.05281   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.624e+01 3.049e+00 -5.328 1.14e-07 \*\*\*  
## fixed.acidity -1.001e-02 1.746e-02 -0.573 0.56658   
## volatile.acidity -1.111e+00 1.203e-01 -9.239 < 2e-16 \*\*\*  
## citric.acid -1.845e-01 1.473e-01 -1.253 0.21053   
## residual.sugar 2.469e-04 1.214e-02 0.020 0.98377   
## chlorides -1.928e+00 4.186e-01 -4.606 4.44e-06 \*\*\*  
## free.sulfur.dioxide 4.672e-03 2.166e-03 2.157 0.03117 \*   
## total.sulfur.dioxide -3.342e-03 7.280e-04 -4.591 4.75e-06 \*\*\*  
## density 2.103e+01 3.426e+00 6.139 1.05e-09 \*\*\*  
## pH -5.857e-01 1.668e-01 -3.510 0.00046 \*\*\*  
## sulphates 8.666e-01 1.111e-01 7.799 1.12e-14 \*\*\*  
## alcohol 3.123e-01 1.760e-02 17.740 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.6485 on 1588 degrees of freedom  
## Multiple R-squared: 0.3722, Adjusted R-squared: 0.3678   
## F-statistic: 85.57 on 11 and 1588 DF, p-value: < 2.2e-16

red.sum.inf

##   
## Call:  
## lm(formula = quality ~ ., data = red.wine.inf)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.68911 -0.36652 -0.04699 0.45202 2.02498   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.197e+01 2.119e+01 1.036 0.3002   
## fixed.acidity 2.499e-02 2.595e-02 0.963 0.3357   
## volatile.acidity -1.084e+00 1.211e-01 -8.948 < 2e-16 \*\*\*  
## citric.acid -1.826e-01 1.472e-01 -1.240 0.2150   
## residual.sugar 1.633e-02 1.500e-02 1.089 0.2765   
## chlorides -1.874e+00 4.193e-01 -4.470 8.37e-06 \*\*\*  
## free.sulfur.dioxide 4.361e-03 2.171e-03 2.009 0.0447 \*   
## total.sulfur.dioxide -3.265e-03 7.287e-04 -4.480 8.00e-06 \*\*\*  
## density -1.788e+01 2.163e+01 -0.827 0.4086   
## pH -4.137e-01 1.916e-01 -2.159 0.0310 \*   
## sulphates 9.163e-01 1.143e-01 8.014 2.13e-15 \*\*\*  
## alcohol 2.762e-01 2.648e-02 10.429 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.648 on 1587 degrees of freedom  
## Multiple R-squared: 0.3606, Adjusted R-squared: 0.3561   
## F-statistic: 81.35 on 11 and 1587 DF, p-value: < 2.2e-16

red.mse

## [1] 0.4173777

red.mse.inf

## [1] 0.4167672

Deleting the point increases the and decreases the mean squared error value by just a little bit, which is not exactly what we want. Hence, we conclude that the points 152, 653, and 1236 are not influential.

As mentioned above, this may be due to trying to fit a discrete quality value to a continuous data.

## Transformation

Previously, the summary of the full model has told us that our model is not doing a good job in fitting the data. And while analyzing the residuals, we realized the fact that fitting a discrete data into a continuous model may be causing such problem.

Before we go into transforamtion on ordinal variables, we will try some conventional transformations.

Log Transformation

First, let’s try a log transformation to see if it will improve our regression.

red.log = lm(log(quality) ~ ., data=red.wine.inf)  
red.log.sum = summary(red.log)  
red.sum.inf

##   
## Call:  
## lm(formula = quality ~ ., data = red.wine.inf)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.68911 -0.36652 -0.04699 0.45202 2.02498   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.197e+01 2.119e+01 1.036 0.3002   
## fixed.acidity 2.499e-02 2.595e-02 0.963 0.3357   
## volatile.acidity -1.084e+00 1.211e-01 -8.948 < 2e-16 \*\*\*  
## citric.acid -1.826e-01 1.472e-01 -1.240 0.2150   
## residual.sugar 1.633e-02 1.500e-02 1.089 0.2765   
## chlorides -1.874e+00 4.193e-01 -4.470 8.37e-06 \*\*\*  
## free.sulfur.dioxide 4.361e-03 2.171e-03 2.009 0.0447 \*   
## total.sulfur.dioxide -3.265e-03 7.287e-04 -4.480 8.00e-06 \*\*\*  
## density -1.788e+01 2.163e+01 -0.827 0.4086   
## pH -4.137e-01 1.916e-01 -2.159 0.0310 \*   
## sulphates 9.163e-01 1.143e-01 8.014 2.13e-15 \*\*\*  
## alcohol 2.762e-01 2.648e-02 10.429 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.648 on 1587 degrees of freedom  
## Multiple R-squared: 0.3606, Adjusted R-squared: 0.3561   
## F-statistic: 81.35 on 11 and 1587 DF, p-value: < 2.2e-16

red.log.sum

##   
## Call:  
## lm(formula = log(quality) ~ ., data = red.wine.inf)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.62878 -0.06002 -0.00361 0.08017 0.33384   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.3093963 3.8764106 0.596 0.5514   
## fixed.acidity 0.0022816 0.0047459 0.481 0.6308   
## volatile.acidity -0.2134468 0.0221490 -9.637 < 2e-16 \*\*\*  
## citric.acid -0.0441610 0.0269180 -1.641 0.1011   
## residual.sugar 0.0014242 0.0027438 0.519 0.6038   
## chlorides -0.3352020 0.0766854 -4.371 1.32e-05 \*\*\*  
## free.sulfur.dioxide 0.0008208 0.0003971 2.067 0.0389 \*   
## total.sulfur.dioxide -0.0005335 0.0001333 -4.003 6.55e-05 \*\*\*  
## density -0.7706234 3.9566152 -0.195 0.8456   
## pH -0.0890447 0.0350425 -2.541 0.0111 \*   
## sulphates 0.1560676 0.0209119 7.463 1.39e-13 \*\*\*  
## alcohol 0.0491903 0.0048438 10.155 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1185 on 1587 degrees of freedom  
## Multiple R-squared: 0.3415, Adjusted R-squared: 0.3369   
## F-statistic: 74.82 on 11 and 1587 DF, p-value: < 2.2e-16

By taking logarthims of quality values, estimates of each of the coefficients changed quite a lot. However, the log transformation decreased the value, which is not good.

Since our goal was to improve the model to better fit the data, we will skip further analysis on the log transformation model and move on.

Square-Root Transformation

Next, we will do a square-root transformation to see if it will improve our regression.

red.sqr = lm(sqrt(quality) ~ ., data=red.wine.inf)  
red.sqr.sum = summary(red.sqr)  
red.sum.inf

##   
## Call:  
## lm(formula = quality ~ ., data = red.wine.inf)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.68911 -0.36652 -0.04699 0.45202 2.02498   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.197e+01 2.119e+01 1.036 0.3002   
## fixed.acidity 2.499e-02 2.595e-02 0.963 0.3357   
## volatile.acidity -1.084e+00 1.211e-01 -8.948 < 2e-16 \*\*\*  
## citric.acid -1.826e-01 1.472e-01 -1.240 0.2150   
## residual.sugar 1.633e-02 1.500e-02 1.089 0.2765   
## chlorides -1.874e+00 4.193e-01 -4.470 8.37e-06 \*\*\*  
## free.sulfur.dioxide 4.361e-03 2.171e-03 2.009 0.0447 \*   
## total.sulfur.dioxide -3.265e-03 7.287e-04 -4.480 8.00e-06 \*\*\*  
## density -1.788e+01 2.163e+01 -0.827 0.4086   
## pH -4.137e-01 1.916e-01 -2.159 0.0310 \*   
## sulphates 9.163e-01 1.143e-01 8.014 2.13e-15 \*\*\*  
## alcohol 2.762e-01 2.648e-02 10.429 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.648 on 1587 degrees of freedom  
## Multiple R-squared: 0.3606, Adjusted R-squared: 0.3561   
## F-statistic: 81.35 on 11 and 1587 DF, p-value: < 2.2e-16

red.sqr.sum

##   
## Call:  
## lm(formula = sqrt(quality) ~ ., data = red.wine.inf)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.64674 -0.07407 -0.00711 0.09766 0.39932   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.4885674 4.5042660 0.997 0.3192   
## fixed.acidity 0.0040616 0.0055146 0.737 0.4615   
## volatile.acidity -0.2395312 0.0257364 -9.307 < 2e-16 \*\*\*  
## citric.acid -0.0451316 0.0312778 -1.443 0.1492   
## residual.sugar 0.0025930 0.0031882 0.813 0.4162   
## chlorides -0.3944442 0.0891060 -4.427 1.02e-05 \*\*\*  
## free.sulfur.dioxide 0.0009465 0.0004614 2.051 0.0404 \*   
## total.sulfur.dioxide -0.0006618 0.0001549 -4.274 2.04e-05 \*\*\*  
## density -2.3940239 4.5974612 -0.521 0.6026   
## pH -0.0953714 0.0407182 -2.342 0.0193 \*   
## sulphates 0.1887500 0.0242990 7.768 1.42e-14 \*\*\*  
## alcohol 0.0581065 0.0056283 10.324 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1377 on 1587 degrees of freedom  
## Multiple R-squared: 0.3524, Adjusted R-squared: 0.348   
## F-statistic: 78.52 on 11 and 1587 DF, p-value: < 2.2e-16

By taking the square-root of quality values, estimates of each of the coefficients changed quite a lot. However, the square-root transformation decreased the value, which is not good.

Since our goal was to improve the model to better fit the data, we will skip further analysis on the square-root transformation model and move on.

Ordinal Variable Regression

Taking the discreteness of the quality values into consideration, we will now look into transforming ordinal variables.

#red.wine$quality = factor(red.wine$quality)  
#red.tran.mdl = polr(quality ~ . , data=red.wine, Hess=TRUE)  
#red.tran.sum = summary(red.tran.mdl)

Still figuring out how to do this part …

## Variable Selection

As we saw in the beginning, some of the variables did not seem significant to our model. Since incorrect model specification can cause serious problems, we will try to find the most appropriate subset of regressors for our model.

We will first define all the possible candidates then compare their adjusted values.

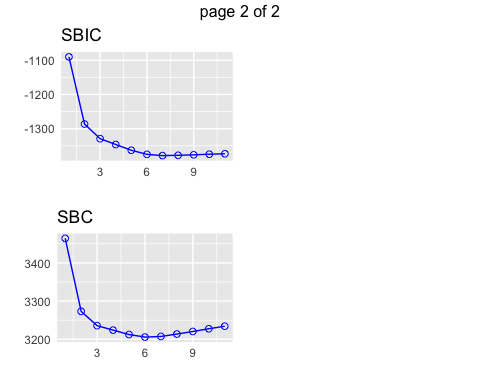
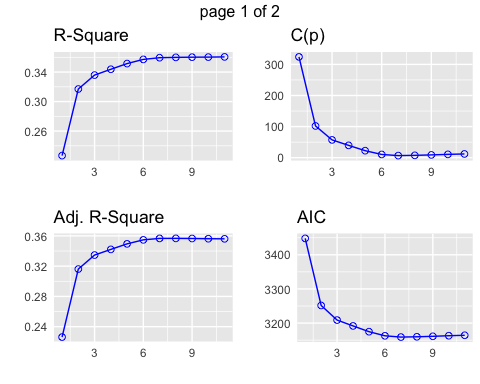
#### Best Subset Regression

First step in choosing the best set of regressors for our model, we will fit all the regression equations involving one, two regressors, and so on. Then we will select the subset of predictors that do the best at meeting some well-defined objective criterion, such as a large adjusted value or the small MSE, Mallow’s , or AIC.

red.ols = ols\_step\_best\_subset(red.mdl.inf)  
red.ols

## Best Subsets Regression   
## --------------------------------------------------------------------------------------------------------------------------------------------------------  
## Model Index Predictors  
## --------------------------------------------------------------------------------------------------------------------------------------------------------  
## 1 alcohol   
## 2 volatile.acidity alcohol   
## 3 volatile.acidity sulphates alcohol   
## 4 volatile.acidity total.sulfur.dioxide sulphates alcohol   
## 5 volatile.acidity chlorides total.sulfur.dioxide sulphates alcohol   
## 6 volatile.acidity chlorides total.sulfur.dioxide pH sulphates alcohol   
## 7 volatile.acidity chlorides free.sulfur.dioxide total.sulfur.dioxide pH sulphates alcohol   
## 8 volatile.acidity citric.acid chlorides free.sulfur.dioxide total.sulfur.dioxide pH sulphates alcohol   
## 9 volatile.acidity citric.acid residual.sugar chlorides free.sulfur.dioxide total.sulfur.dioxide pH sulphates alcohol   
## 10 fixed.acidity volatile.acidity citric.acid residual.sugar chlorides free.sulfur.dioxide total.sulfur.dioxide pH sulphates alcohol   
## 11 fixed.acidity volatile.acidity citric.acid residual.sugar chlorides free.sulfur.dioxide total.sulfur.dioxide density pH sulphates alcohol   
## --------------------------------------------------------------------------------------------------------------------------------------------------------  
##   
## Subsets Regression Summary   
## --------------------------------------------------------------------------------------------------------------------------------------  
## Adj. Pred   
## Model R-Square R-Square R-Square C(p) AIC SBIC SBC MSEP FPE HSP APC   
## --------------------------------------------------------------------------------------------------------------------------------------  
## 1 0.2267 0.2263 0.2246 324.1115 3448.1135 -1090.3748 3464.2449 806.8797 0.5052 3e-04 0.7752   
## 2 0.3170 0.3161 0.3139 102.0818 3251.6275 -1286.4844 3273.1360 713.1345 0.4468 3e-04 0.6856   
## 3 0.3359 0.3346 0.3316 57.1879 3208.7683 -1329.2376 3235.6540 693.8409 0.4350 3e-04 0.6674   
## 4 0.3438 0.3421 0.3386 39.6184 3191.6693 -1346.2787 3223.9321 686.0331 0.4304 3e-04 0.6603   
## 5 0.3515 0.3495 0.3454 22.4791 3174.7667 -1363.0770 3212.4066 678.3967 0.4259 3e-04 0.6534   
## 6 0.3572 0.3548 0.3501 10.3837 3162.7015 -1375.0322 3205.7185 672.8782 0.4227 3e-04 0.6485   
## 7 0.3595 0.3567 0.3515 6.6823 3158.9769 -1378.6948 3207.3711 670.8952 0.4217 3e-04 0.6470   
## 8 0.3599 0.3567 0.351 7.5516 3159.8391 -1377.8080 3213.6105 670.8399 0.4219 3e-04 0.6473   
## 9 0.3602 0.3565 0.3501 8.9404 3161.2237 -1376.4025 3220.3722 671.0041 0.4223 3e-04 0.6479   
## 10 0.3603 0.3562 0.3491 10.6832 3162.9648 -1374.6439 3227.4904 671.3182 0.4227 3e-04 0.6486   
## 11 0.3606 0.3561 0.3483 12.0000 3164.2766 -1373.3075 3234.1793 671.4524 0.4231 3e-04 0.6491   
## --------------------------------------------------------------------------------------------------------------------------------------  
## AIC: Akaike Information Criteria   
## SBIC: Sawa's Bayesian Information Criteria   
## SBC: Schwarz Bayesian Criteria   
## MSEP: Estimated error of prediction, assuming multivariate normality   
## FPE: Final Prediction Error   
## HSP: Hocking's Sp   
## APC: Amemiya Prediction Criteria

plot(red.ols)



best.red = lm(quality ~ volatile.acidity + chlorides + free.sulfur.dioxide + total.sulfur.dioxide + pH + sulphates + alcohol, data=red.wine.inf)  
best.red.sum = summary(best.red)

Looking at the plots above, the increase in and Adjusted almost flattens by model 7. In fact, change in Mallow’s statistic and AIC drastically decreases at model 7 as well.

Hence, best subset regression tells us that model 7, which has the following variables:

* volatile acidity
* chlorides
* free sulfur dioxide
* total sulfur dioxide
* pH
* sulphates
* alcohol

best meet our criterion, which perfectly aligns with our expectation we had when we saw the large P-values in the full model.

#### Stepwise Regression Methods

Since comparing all possible regressions and the best subsets take a lot of time and work, we will choose only a few models by adding or deleting regressors one at a time. Here, keep in mind that we used AIC values to compare each steps.

Forward Selection

Start with a model with no regressors included, then we will add regressors as we go on.

start.mdl = lm(quality~1, data=red.wine.inf)  
scope.mdl = lm(quality~., data=red.wine.inf)  
step(start.mdl, direction = "forward", scope=formula(scope.mdl))

## Start: AIC=-682.5  
## quality ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + alcohol 1 236.295 805.87 -1091.65  
## + volatile.acidity 1 158.967 883.20 -945.14  
## + sulphates 1 65.865 976.30 -784.89  
## + citric.acid 1 53.405 988.76 -764.61  
## + total.sulfur.dioxide 1 35.707 1006.46 -736.24  
## + density 1 31.887 1010.28 -730.19  
## + chlorides 1 17.318 1024.85 -707.29  
## + fixed.acidity 1 16.038 1026.13 -705.29  
## + pH 1 3.473 1038.69 -685.84  
## + free.sulfur.dioxide 1 2.674 1039.49 -684.61  
## <none> 1042.17 -682.50  
## + residual.sugar 1 0.197 1041.97 -680.80  
##   
## Step: AIC=-1091.65  
## quality ~ alcohol  
##   
## Df Sum of Sq RSS AIC  
## + volatile.acidity 1 94.074 711.80 -1288.1  
## + sulphates 1 44.977 760.89 -1181.5  
## + citric.acid 1 31.953 773.92 -1154.3  
## + pH 1 26.362 779.51 -1142.8  
## + fixed.acidity 1 24.623 781.25 -1139.3  
## + total.sulfur.dioxide 1 8.270 797.60 -1106.2  
## + density 1 5.203 800.67 -1100.0  
## <none> 805.87 -1091.7  
## + chlorides 1 0.611 805.26 -1090.9  
## + free.sulfur.dioxide 1 0.325 805.55 -1090.3  
## + residual.sugar 1 0.041 805.83 -1089.7  
##   
## Step: AIC=-1288.14  
## quality ~ alcohol + volatile.acidity  
##   
## Df Sum of Sq RSS AIC  
## + sulphates 1 19.6916 692.10 -1331.0  
## + total.sulfur.dioxide 1 6.3730 705.42 -1300.5  
## + pH 1 5.9515 705.84 -1299.6  
## + fixed.acidity 1 5.7061 706.09 -1299.0  
## + density 1 1.9410 709.86 -1290.5  
## <none> 711.80 -1288.1  
## + free.sulfur.dioxide 1 0.6621 711.13 -1287.6  
## + chlorides 1 0.3762 711.42 -1287.0  
## + citric.acid 1 0.1936 711.60 -1286.6  
## + residual.sugar 1 0.0101 711.79 -1286.2  
##   
## Step: AIC=-1331  
## quality ~ alcohol + volatile.acidity + sulphates  
##   
## Df Sum of Sq RSS AIC  
## + total.sulfur.dioxide 1 8.2176 683.89 -1348.1  
## + chlorides 1 7.4925 684.61 -1346.4  
## + fixed.acidity 1 3.3282 688.78 -1336.7  
## + pH 1 3.0454 689.06 -1336.0  
## + free.sulfur.dioxide 1 1.1129 690.99 -1331.6  
## <none> 692.10 -1331.0  
## + citric.acid 1 0.2522 691.85 -1329.6  
## + density 1 0.2222 691.88 -1329.5  
## + residual.sugar 1 0.0143 692.09 -1329.0  
##   
## Step: AIC=-1348.1  
## quality ~ alcohol + volatile.acidity + sulphates + total.sulfur.dioxide  
##   
## Df Sum of Sq RSS AIC  
## + chlorides 1 8.0370 675.85 -1365.0  
## + pH 1 3.3094 680.58 -1353.8  
## + fixed.acidity 1 2.1037 681.78 -1351.0  
## + free.sulfur.dioxide 1 1.3557 682.53 -1349.3  
## <none> 683.89 -1348.1  
## + residual.sugar 1 0.2634 683.62 -1346.7  
## + density 1 0.1077 683.78 -1346.3  
## + citric.acid 1 0.0730 683.81 -1346.3  
##   
## Step: AIC=-1365  
## quality ~ alcohol + volatile.acidity + sulphates + total.sulfur.dioxide +   
## chlorides  
##   
## Df Sum of Sq RSS AIC  
## + pH 1 5.9189 669.93 -1377.1  
## + fixed.acidity 1 2.4065 673.44 -1368.7  
## + free.sulfur.dioxide 1 1.2403 674.61 -1365.9  
## <none> 675.85 -1365.0  
## + residual.sugar 1 0.5531 675.30 -1364.3  
## + citric.acid 1 0.1615 675.69 -1363.4  
## + density 1 0.1526 675.70 -1363.4  
##   
## Step: AIC=-1377.06  
## quality ~ alcohol + volatile.acidity + sulphates + total.sulfur.dioxide +   
## chlorides + pH  
##   
## Df Sum of Sq RSS AIC  
## + free.sulfur.dioxide 1 2.39413 667.54 -1380.8  
## <none> 669.93 -1377.1  
## + citric.acid 1 0.80525 669.13 -1377.0  
## + residual.sugar 1 0.28390 669.65 -1375.7  
## + density 1 0.04468 669.89 -1375.2  
## + fixed.acidity 1 0.01040 669.92 -1375.1  
##   
## Step: AIC=-1380.79  
## quality ~ alcohol + volatile.acidity + sulphates + total.sulfur.dioxide +   
## chlorides + pH + free.sulfur.dioxide  
##   
## Df Sum of Sq RSS AIC  
## <none> 667.54 -1380.8  
## + citric.acid 1 0.47480 667.06 -1379.9  
## + residual.sugar 1 0.16673 667.37 -1379.2  
## + density 1 0.03079 667.51 -1378.9  
## + fixed.acidity 1 0.00663 667.53 -1378.8

##   
## Call:  
## lm(formula = quality ~ alcohol + volatile.acidity + sulphates +   
## total.sulfur.dioxide + chlorides + pH + free.sulfur.dioxide,   
## data = red.wine.inf)  
##   
## Coefficients:  
## (Intercept) alcohol volatile.acidity   
## 4.430099 0.289303 -1.012753   
## sulphates total.sulfur.dioxide chlorides   
## 0.882665 -0.003482 -2.017814   
## pH free.sulfur.dioxide   
## -0.482661 0.005077

forward.red= lm(quality ~ alcohol + volatile.acidity + sulphates +   
 total.sulfur.dioxide + chlorides + pH + free.sulfur.dioxide,   
 data = red.wine.inf)  
forward.red.sum = summary(forward.red)

Our final model has 7 of the 11 possible predictors (written in the order they were added):

* alcohol
* volatile acidity
* sulphates
* total sulfur dioxide
* chlorides
* pH
* free sulfur dioxide

Coincidentally, the chosen 7 variables are identical as our choice for best subsets regression.

Backward Elimination

Start with a model with all the regressors included, then we will eliminate the insignificant regressors as we go on.

start.mdl2 = lm(quality~., data=red.wine.inf)  
step(start.mdl2, direction = "backward")

## Start: AIC=-1375.49  
## quality ~ fixed.acidity + volatile.acidity + citric.acid + residual.sugar +   
## chlorides + free.sulfur.dioxide + total.sulfur.dioxide +   
## density + pH + sulphates + alcohol  
##   
## Df Sum of Sq RSS AIC  
## - density 1 0.287 666.70 -1376.8  
## - fixed.acidity 1 0.389 666.80 -1376.5  
## - residual.sugar 1 0.498 666.91 -1376.3  
## - citric.acid 1 0.646 667.06 -1375.9  
## <none> 666.41 -1375.5  
## - free.sulfur.dioxide 1 1.694 668.10 -1373.4  
## - pH 1 1.957 668.37 -1372.8  
## - chlorides 1 8.391 674.80 -1357.5  
## - total.sulfur.dioxide 1 8.427 674.84 -1357.4  
## - sulphates 1 26.971 693.38 -1314.0  
## - volatile.acidity 1 33.620 700.03 -1298.8  
## - alcohol 1 45.672 712.08 -1271.5  
##   
## Step: AIC=-1376.8  
## quality ~ fixed.acidity + volatile.acidity + citric.acid + residual.sugar +   
## chlorides + free.sulfur.dioxide + total.sulfur.dioxide +   
## pH + sulphates + alcohol  
##   
## Df Sum of Sq RSS AIC  
## - fixed.acidity 1 0.108 666.81 -1378.5  
## - residual.sugar 1 0.231 666.93 -1378.2  
## - citric.acid 1 0.654 667.35 -1377.2  
## <none> 666.70 -1376.8  
## - free.sulfur.dioxide 1 1.829 668.53 -1374.4  
## - pH 1 4.325 671.02 -1368.5  
## - total.sulfur.dioxide 1 8.728 675.43 -1358.0  
## - chlorides 1 8.761 675.46 -1357.9  
## - sulphates 1 27.287 693.98 -1314.7  
## - volatile.acidity 1 35.000 701.70 -1297.0  
## - alcohol 1 119.669 786.37 -1114.8  
##   
## Step: AIC=-1378.54  
## quality ~ volatile.acidity + citric.acid + residual.sugar + chlorides +   
## free.sulfur.dioxide + total.sulfur.dioxide + pH + sulphates +   
## alcohol  
##   
## Df Sum of Sq RSS AIC  
## - residual.sugar 1 0.257 667.06 -1379.9  
## - citric.acid 1 0.565 667.37 -1379.2  
## <none> 666.81 -1378.5  
## - free.sulfur.dioxide 1 1.901 668.71 -1376.0  
## - pH 1 7.065 673.87 -1363.7  
## - chlorides 1 9.940 676.75 -1356.9  
## - total.sulfur.dioxide 1 10.031 676.84 -1356.7  
## - sulphates 1 27.673 694.48 -1315.5  
## - volatile.acidity 1 36.234 703.04 -1295.9  
## - alcohol 1 120.633 787.44 -1114.7  
##   
## Step: AIC=-1379.93  
## quality ~ volatile.acidity + citric.acid + chlorides + free.sulfur.dioxide +   
## total.sulfur.dioxide + pH + sulphates + alcohol  
##   
## Df Sum of Sq RSS AIC  
## - citric.acid 1 0.475 667.54 -1380.8  
## <none> 667.06 -1379.9  
## - free.sulfur.dioxide 1 2.064 669.13 -1377.0  
## - pH 1 7.138 674.20 -1364.9  
## - total.sulfur.dioxide 1 9.828 676.89 -1358.5  
## - chlorides 1 9.832 676.89 -1358.5  
## - sulphates 1 27.446 694.51 -1317.5  
## - volatile.acidity 1 35.977 703.04 -1297.9  
## - alcohol 1 122.667 789.73 -1112.0  
##   
## Step: AIC=-1380.79  
## quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +   
## total.sulfur.dioxide + pH + sulphates + alcohol  
##   
## Df Sum of Sq RSS AIC  
## <none> 667.54 -1380.8  
## - free.sulfur.dioxide 1 2.394 669.93 -1377.1  
## - pH 1 7.073 674.61 -1365.9  
## - total.sulfur.dioxide 1 10.787 678.32 -1357.2  
## - chlorides 1 10.809 678.35 -1357.1  
## - sulphates 1 27.060 694.60 -1319.2  
## - volatile.acidity 1 42.318 709.85 -1284.5  
## - alcohol 1 124.483 792.02 -1109.4

##   
## Call:  
## lm(formula = quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +   
## total.sulfur.dioxide + pH + sulphates + alcohol, data = red.wine.inf)  
##   
## Coefficients:  
## (Intercept) volatile.acidity chlorides   
## 4.430099 -1.012753 -2.017814   
## free.sulfur.dioxide total.sulfur.dioxide pH   
## 0.005077 -0.003482 -0.482661   
## sulphates alcohol   
## 0.882665 0.289303

backward.red = lm(quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +   
 total.sulfur.dioxide + pH + sulphates + alcohol, data = red.wine.inf)  
backward.red.sum = summary(backward.red)

Our final model has 7 of the 11 possible predictors (written in their order in the model):

* volatile acidity
* chlorides
* free sulfur dioxide
* total sulfur dioxide
* pH
* sulphates
* alcohol

Again, the chosen 7 variables are identical as our choice for best subsets regression and forward selection.

Stepwise Regression

Repeat forward selection and backward elimination.

step(red.mdl.inf, direction= "both")

## Start: AIC=-1375.49  
## quality ~ fixed.acidity + volatile.acidity + citric.acid + residual.sugar +   
## chlorides + free.sulfur.dioxide + total.sulfur.dioxide +   
## density + pH + sulphates + alcohol  
##   
## Df Sum of Sq RSS AIC  
## - density 1 0.287 666.70 -1376.8  
## - fixed.acidity 1 0.389 666.80 -1376.5  
## - residual.sugar 1 0.498 666.91 -1376.3  
## - citric.acid 1 0.646 667.06 -1375.9  
## <none> 666.41 -1375.5  
## - free.sulfur.dioxide 1 1.694 668.10 -1373.4  
## - pH 1 1.957 668.37 -1372.8  
## - chlorides 1 8.391 674.80 -1357.5  
## - total.sulfur.dioxide 1 8.427 674.84 -1357.4  
## - sulphates 1 26.971 693.38 -1314.0  
## - volatile.acidity 1 33.620 700.03 -1298.8  
## - alcohol 1 45.672 712.08 -1271.5  
##   
## Step: AIC=-1376.8  
## quality ~ fixed.acidity + volatile.acidity + citric.acid + residual.sugar +   
## chlorides + free.sulfur.dioxide + total.sulfur.dioxide +   
## pH + sulphates + alcohol  
##   
## Df Sum of Sq RSS AIC  
## - fixed.acidity 1 0.108 666.81 -1378.5  
## - residual.sugar 1 0.231 666.93 -1378.2  
## - citric.acid 1 0.654 667.35 -1377.2  
## <none> 666.70 -1376.8  
## + density 1 0.287 666.41 -1375.5  
## - free.sulfur.dioxide 1 1.829 668.53 -1374.4  
## - pH 1 4.325 671.02 -1368.5  
## - total.sulfur.dioxide 1 8.728 675.43 -1358.0  
## - chlorides 1 8.761 675.46 -1357.9  
## - sulphates 1 27.287 693.98 -1314.7  
## - volatile.acidity 1 35.000 701.70 -1297.0  
## - alcohol 1 119.669 786.37 -1114.8  
##   
## Step: AIC=-1378.54  
## quality ~ volatile.acidity + citric.acid + residual.sugar + chlorides +   
## free.sulfur.dioxide + total.sulfur.dioxide + pH + sulphates +   
## alcohol  
##   
## Df Sum of Sq RSS AIC  
## - residual.sugar 1 0.257 667.06 -1379.9  
## - citric.acid 1 0.565 667.37 -1379.2  
## <none> 666.81 -1378.5  
## + fixed.acidity 1 0.108 666.70 -1376.8  
## + density 1 0.005 666.80 -1376.5  
## - free.sulfur.dioxide 1 1.901 668.71 -1376.0  
## - pH 1 7.065 673.87 -1363.7  
## - chlorides 1 9.940 676.75 -1356.9  
## - total.sulfur.dioxide 1 10.031 676.84 -1356.7  
## - sulphates 1 27.673 694.48 -1315.5  
## - volatile.acidity 1 36.234 703.04 -1295.9  
## - alcohol 1 120.633 787.44 -1114.7  
##   
## Step: AIC=-1379.93  
## quality ~ volatile.acidity + citric.acid + chlorides + free.sulfur.dioxide +   
## total.sulfur.dioxide + pH + sulphates + alcohol  
##   
## Df Sum of Sq RSS AIC  
## - citric.acid 1 0.475 667.54 -1380.8  
## <none> 667.06 -1379.9  
## + residual.sugar 1 0.257 666.81 -1378.5  
## + fixed.acidity 1 0.133 666.93 -1378.2  
## + density 1 0.028 667.03 -1378.0  
## - free.sulfur.dioxide 1 2.064 669.13 -1377.0  
## - pH 1 7.138 674.20 -1364.9  
## - total.sulfur.dioxide 1 9.828 676.89 -1358.5  
## - chlorides 1 9.832 676.89 -1358.5  
## - sulphates 1 27.446 694.51 -1317.5  
## - volatile.acidity 1 35.977 703.04 -1297.9  
## - alcohol 1 122.667 789.73 -1112.0  
##   
## Step: AIC=-1380.79  
## quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +   
## total.sulfur.dioxide + pH + sulphates + alcohol  
##   
## Df Sum of Sq RSS AIC  
## <none> 667.54 -1380.8  
## + citric.acid 1 0.475 667.06 -1379.9  
## + residual.sugar 1 0.167 667.37 -1379.2  
## + density 1 0.031 667.51 -1378.9  
## + fixed.acidity 1 0.007 667.53 -1378.8  
## - free.sulfur.dioxide 1 2.394 669.93 -1377.1  
## - pH 1 7.073 674.61 -1365.9  
## - total.sulfur.dioxide 1 10.787 678.32 -1357.2  
## - chlorides 1 10.809 678.35 -1357.1  
## - sulphates 1 27.060 694.60 -1319.2  
## - volatile.acidity 1 42.318 709.85 -1284.5  
## - alcohol 1 124.483 792.02 -1109.4

##   
## Call:  
## lm(formula = quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +   
## total.sulfur.dioxide + pH + sulphates + alcohol, data = red.wine.inf)  
##   
## Coefficients:  
## (Intercept) volatile.acidity chlorides   
## 4.430099 -1.012753 -2.017814   
## free.sulfur.dioxide total.sulfur.dioxide pH   
## 0.005077 -0.003482 -0.482661   
## sulphates alcohol   
## 0.882665 0.289303

both.red = lm(quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +   
 total.sulfur.dioxide + pH + sulphates + alcohol, data = red.wine.inf)  
both.red.sum = summary(both.red)

Our final model has 7 of the 11 possible predictors (written in their order in the model):

* volatile acidity
* chlorides
* free sulfur dioxide
* total sulfur dioxide
* pH
* sulphates
* alcohol

As it turns out, the chosen 7 variables are identical for all our variable selection methods.

Now, let’s compare all our models by their adjusted .

# Adjusted R-squared for the full model  
red.rsq = red.sum.inf$adj.r.squared  
red.rsq

## [1] 0.3561195

# Adjusted R-squared for best subset regression model  
best.red.rsq = best.red.sum$adj.r.squared  
best.red.rsq

## [1] 0.3566527

# Adjusted R-squared for forward selection model  
forward.red.rsq = forward.red.sum$adj.r.squared  
forward.red.rsq

## [1] 0.3566527

# Adjusted R-squared for backward elimination model  
backward.red.rsq = backward.red.sum$adj.r.squared  
backward.red.rsq

## [1] 0.3566527

# Adjusted R-squared for stepwise regression model  
both.red.rsq = both.red.sum$adj.r.squared  
both.red.rsq

## [1] 0.3566527

Clearly, even after selecting the appropriate regressors, the adjusted did not improve. In other words, the subset models do not show much of a higher performance. In fact, as we saw above, all the variable selection models choose the same 7 variables and even generates equal adjusted values.

Hence, from now on, the best model used will be

Quality = + \* volatile acidity + \* chlorides + \* free sulfur dioxide + \* total sulfur dioxide + \* pH + \* sulphates + \* alcohol.

However, although these are all the variable selection method we learned, we should always keep in mind that there will be several equally good models. That is, we might be ignorant of the background knowledge of the collected data, i.e. there may be a whole other variable that affects wine quality.

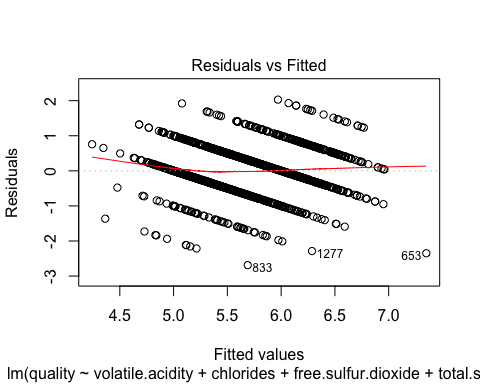
With that in mind, we will assess our final model by investigating the residual plots.

#### Residual Analysis

fin.red = lm(quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +   
 total.sulfur.dioxide + pH + sulphates + alcohol, data = red.wine.inf)

##### Residual vs. Fitted

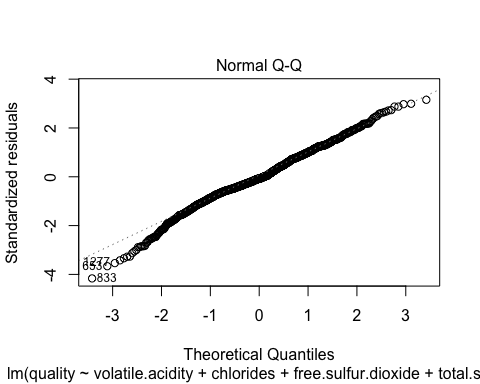
plot(fin.red, which=1)



Comments on the residual vs. fitted plot

##### Normal Q-Q

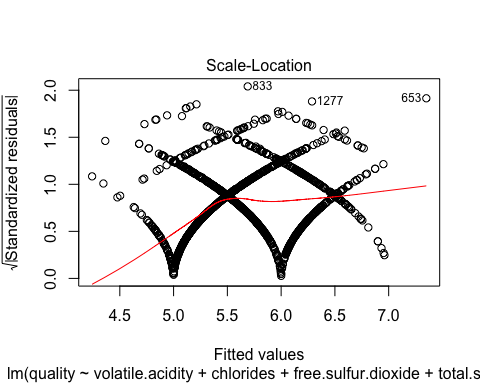
plot(fin.red, which=2)



We observe that the residuals for our model meet the normality assumption. The negative skew that we saw in the full model has reduced, which is a good thing.

##### Scale-Location

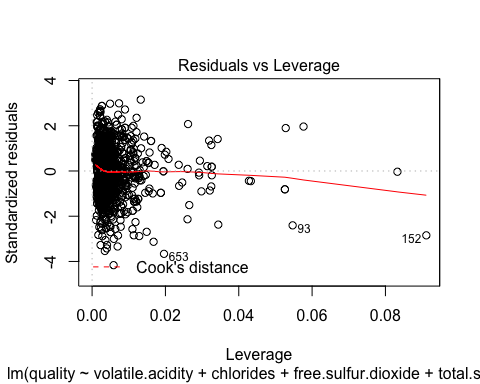
plot(fin.red, which=3)



Comments on the Scale-Location Plot

##### Residuals vs. Leverage

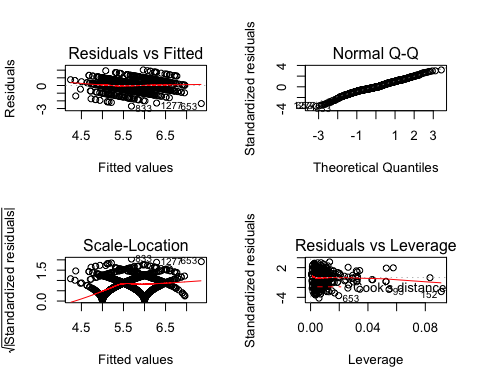
plot(fin.red, which=5)



Comments on the Residuals vs. Leverage Plot

##### Overall Residual Plots

par(mfrow = c(2,2))  
plot(fin.red, which=1)  
plot(fin.red, which=2)  
plot(fin.red, which=3)  
plot(fin.red, which=5)

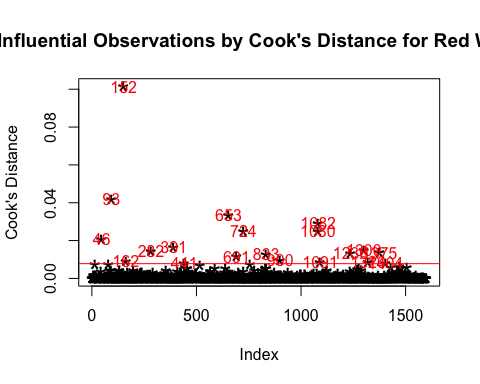


Cook’s Distance

red.cooksd2 = cooks.distance(fin.red)  
max.di2 = max(red.cooksd2)  
max.di2

## [1] 0.101497

plot(red.cooksd2, pch="\*", cex=2, ylab = "Cook's Distance", main="Influential Observations by Cook's Distance for Red Wine")   
abline(h = 10\*mean(red.cooksd2, na.rm=T), col="red") # adding the cutoff line  
text(x=1:length(red.cooksd2)+1, y=red.cooksd2, labels=ifelse(red.cooksd2>10\*mean(red.cooksd2, na.rm=T),names(red.cooksd2),""), col="red")

 ## Model Validation

Before we jump to any conclusions, we should check the validity of our model. There is a good chance that this data was made so that one can predict the quality of a wine based on specific chemical attributes.

We will perform cross validation in order to see how well our model predicts the quality of red wine. We do this by dividing the dataset in such a way that 80 percent of the dataset is part of the training set and 20 percent of the dataset is the testing set.

Then we will compare the actual value to the predicted value. We repeat the steps of validation 5 times to check the values of R squared, root mean square error and mean absolute error, which will tell us how the model behaves.

Cross Validation - Full Model

First, we will see the prediction power of our full model with all our regressors.

set.seed(71168)  
sample.n = ceiling(0.8\*length(red.wine.inf$quality))  
  
par(mfrow = c(3,3))  
for(i in 1:5){  
 train.sample = sample(c(1:length(red.wine.inf$quality)),sample.n)  
 train.sample = sort(train.sample)  
   
 train.data = red.wine.inf[train.sample,]  
 test.data = red.wine.inf[-train.sample,]  
   
 train.mdl = lm(train.data$quality~., data = train.data)  
 summary(train.mdl)   
   
 preds = predict(train.mdl,test.data)  
   
 plot(test.data$quality,preds,xlim=c(4,10),ylim=c(4,10))  
 abline(c(0,1))  
   
 # R-squared  
 R.sq = R2(preds, test.data$quality)  
   
 #RMSPE  
 RMSPE = RMSE(preds, test.data$quality)  
   
 #MAPE  
 MAPE = MAE(preds, test.data$quality)  
   
 print(c(i,R.sq,RMSPE,MAPE))  
}

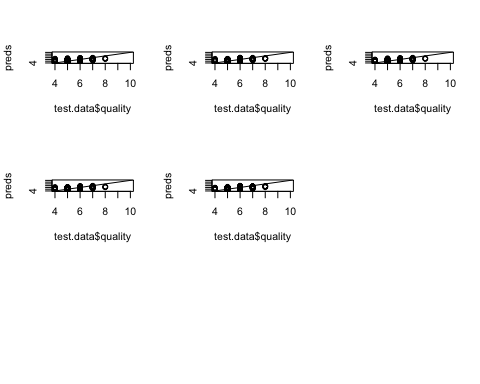
## [1] 1.0000000 0.3304176 0.6523209 0.5083334

## [1] 2.0000000 0.2971903 0.6899844 0.5340842

## [1] 3.0000000 0.3755500 0.6213802 0.4892158

## [1] 4.0000000 0.3630819 0.6464978 0.4995009

## [1] 5.0000000 0.3820851 0.6150485 0.4854226



Here are the things that we notice from our numerical output.

* value remains to be small. An ideal model explains at least 80% of the data, while only 38.4% of our model could be explained.
* Root mean square error (or MSPE), the difference between the predicted quality and test quality, is pretty high. Our goal is to have as small a difference as possible, but in our case, it goes up to 0.65.
* Mean square error (or MSE) is also pretty high, which is not good.

Now, here are the things that we see from our graphical observations.

* When predicting quality 4, the model tends to overestimate the value.
* When predicting quality 7 and 8, the model tends to underestimate the value.
* Our model does predict quality 5 and 6, the huge variability around the value implies that many points are further away from the centre.

Cross Validation - Final Model

Next, let’s take a look at the prediction power of our final model with selected regressors.

set.seed(71168)  
sample.n = ceiling(0.8\*length(red.wine.inf$quality))  
  
par(mfrow = c(3,3))  
  
for(i in 1:5){  
 train.sample = sample(c(1:length(red.wine.inf$quality)),sample.n)  
 train.sample = sort(train.sample)  
   
 train.data = red.wine.inf[train.sample,]  
 test.data = red.wine.inf[-train.sample,]  
   
 train.fin = lm(quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +   
 total.sulfur.dioxide + pH + sulphates + alcohol, data = train.data)  
 summary(train.fin)   
   
 preds.fin = predict(train.fin,test.data)  
   
 plot(test.data$quality,preds.fin,xlim=c(4,10),ylim=c(4,10))  
 abline(c(0,1))  
   
 # R-squared  
 R.sq = R2(preds.fin, test.data$quality)  
   
 #RMSPE  
 RMSPE = RMSE(preds.fin, test.data$quality)  
   
 #MAPE  
 MAPE = MAE(preds.fin, test.data$quality)  
   
 print(c(i,R.sq,RMSPE,MAPE))  
}

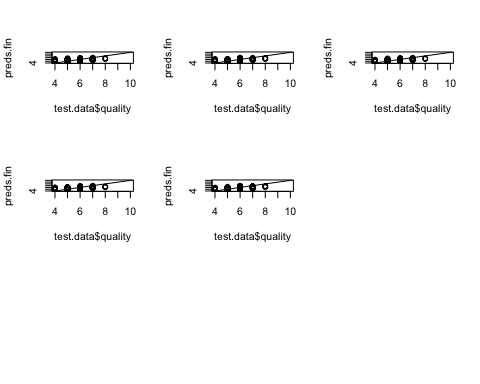
## [1] 1.0000000 0.3424918 0.6464932 0.5066120

## [1] 2.0000000 0.2990017 0.6893982 0.5350565

## [1] 3.0000000 0.3875867 0.6154867 0.4836097

## [1] 4.0000000 0.3754524 0.6405169 0.4960281

## [1] 5.0000000 0.3811559 0.6155229 0.4847436



Both the numerical and graphical observations do not have a lot of difference from our full model.

Hence we conclude that both our models have no prediction powers.

# Conclusion

Because the dataset only contains physiochemical variables assosciated with the wine, we do not have the full picture. Additional variables could help us paint a better picture of how the quality of the wine is affected.

For example, take the grape. What is the type of grape? What region is it from? What is the quality or pH of the soil it was grown in? How does the quality of grape compare to past harvests?

What was the length of fermentation? What was the average temperature during the fermentation process? What was the size of the batch?

All of this additional information could help us to understand better the resulting effect on the quality of wine. As shown from our analysis, we unfortunately can’t come up with a model that accurately predicts the quality of wine given only the variables provided.

# References

Here are the works cited.

Montgomery, D. C., Peck, E. A., & Vining, G. G. (2012). *Introduction to Linear Regression Analysis - 5th ed.* Hoboken, New Jersey: John Wiley & Sons, Inc.

P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. *Modeling wine preferences by data mining from physicochemical properties.* In Decision Support Systems, Elsevier, 47(4):547-553, 2009.