HW3

Valerie Mac

February 16, 2017

You are now the data slave to the principal investigator Dr. Vinca Monster. Dr. M is in the Grape Program at State U, and you are just a poor graduate student trying to get your degree. Dr. M is interested in wine preferences and the influences of physico-chemical properties on preferences. Her laboratory has gathered an extensive dataset on Portugese white varietals.

You will find the white\_wines.csv file and its description in my github repo (<https://github.com/vhertzb/Regression-1>)[<https://github.com/vhertzb/Regression-1>].

Please use the techniques you have learned in the last two classes, specifically exploratory data analysis and linear regression, to determine association of the wine properties on preference.

Prepare a report for presentation at the next Monster lab meeting about this dataset.

Rubric:

Exploration (summary statistics (the m's), univariate graphs, multivariate graphs) Regression (Models explored, diagnostics completed, final model choice, justification)

Please include a concluding paragraph (or two) about the implications of your findings.

See what is in the data set

# see what is in the ww (White\_wines.csv) dataset  
summary(ww)

## fixed.acidity volatile.acidity citric.acid residual.sugar   
## Min. : 3.800 Min. :0.0800 Min. :0.0000 Min. : 0.600   
## 1st Qu.: 6.300 1st Qu.:0.2100 1st Qu.:0.2700 1st Qu.: 1.700   
## Median : 6.800 Median :0.2600 Median :0.3200 Median : 5.200   
## Mean : 6.855 Mean :0.2782 Mean :0.3342 Mean : 6.391   
## 3rd Qu.: 7.300 3rd Qu.:0.3200 3rd Qu.:0.3900 3rd Qu.: 9.900   
## Max. :14.200 Max. :1.1000 Max. :1.6600 Max. :65.800   
## chlorides free.sulfur.dioxide total.sulfur.dioxide  
## Min. :0.00900 Min. : 2.00 Min. : 9.0   
## 1st Qu.:0.03600 1st Qu.: 23.00 1st Qu.:108.0   
## Median :0.04300 Median : 34.00 Median :134.0   
## Mean :0.04577 Mean : 35.31 Mean :138.4   
## 3rd Qu.:0.05000 3rd Qu.: 46.00 3rd Qu.:167.0   
## Max. :0.34600 Max. :289.00 Max. :440.0   
## density pH sulphates alcohol   
## Min. :0.9871 Min. :2.720 Min. :0.2200 Min. : 8.00   
## 1st Qu.:0.9917 1st Qu.:3.090 1st Qu.:0.4100 1st Qu.: 9.50   
## Median :0.9937 Median :3.180 Median :0.4700 Median :10.40   
## Mean :0.9940 Mean :3.188 Mean :0.4898 Mean :10.51   
## 3rd Qu.:0.9961 3rd Qu.:3.280 3rd Qu.:0.5500 3rd Qu.:11.40   
## Max. :1.0390 Max. :3.820 Max. :1.0800 Max. :14.20   
## quality   
## Min. :3.000   
## 1st Qu.:5.000   
## Median :6.000   
## Mean :5.878   
## 3rd Qu.:6.000   
## Max. :9.000

data()

#load up necessary packages  
library(tidyverse)

## Loading tidyverse: ggplot2  
## Loading tidyverse: tibble  
## Loading tidyverse: tidyr  
## Loading tidyverse: readr  
## Loading tidyverse: purrr  
## Loading tidyverse: dplyr

## Conflicts with tidy packages ----------------------------------------------

## filter(): dplyr, stats  
## lag(): dplyr, stats

library(car)

##   
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':  
##   
## recode

## The following object is masked from 'package:purrr':  
##   
## some

library(stargazer)

##   
## Please cite as:

## Hlavac, Marek (2015). stargazer: Well-Formatted Regression and Summary Statistics Tables.

## R package version 5.2. http://CRAN.R-project.org/package=stargazer

# let's add more descriptive stats to our table  
# this means we now have more columns - one per stat  
#for this chunk, I was unable to get the prett descriptives table to work - see error below, but at least I have some descriptive data from the previous step  
#ww %>%  
 #summarise(nQuality = length(quality),  
 #minQuality = min(quality),  
 #meanQuality = mean(quality),  
 #sdQuality = sd(quality),  
 #pct25= quantile(quality, 0.25),  
 #medianQuality = median(quality),  
 #maxQuality = max(quality)) %>%  
 # knitr::kable(col.names=c("N","min",  
 #"mean","sd","median","max"),  
 #caption="Quality Rating: Descriptive Stats")

# Developing the model on ~90% of the dataset, then test it on the remaining ~10% of the data. In the datascience world, we call that first step "training" the model.

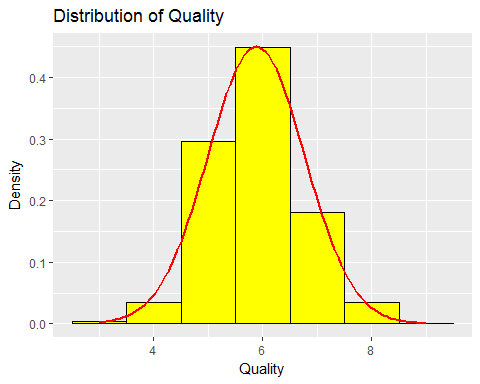
# divide the dataset into a training and a testing set based on a random uniform number on fixed seed - using a diffferent set of subjects to train it than what you are testing it with- use the seed for reproducible results - ww$quality is adding an additonal variable to the temporary files with random numbers  
set.seed(20170214)  
ww$group <- runif(length(ww$quality), min = 0, max = 1)  
summary(ww)

## fixed.acidity volatile.acidity citric.acid residual.sugar   
## Min. : 3.800 Min. :0.0800 Min. :0.0000 Min. : 0.600   
## 1st Qu.: 6.300 1st Qu.:0.2100 1st Qu.:0.2700 1st Qu.: 1.700   
## Median : 6.800 Median :0.2600 Median :0.3200 Median : 5.200   
## Mean : 6.855 Mean :0.2782 Mean :0.3342 Mean : 6.391   
## 3rd Qu.: 7.300 3rd Qu.:0.3200 3rd Qu.:0.3900 3rd Qu.: 9.900   
## Max. :14.200 Max. :1.1000 Max. :1.6600 Max. :65.800   
## chlorides free.sulfur.dioxide total.sulfur.dioxide  
## Min. :0.00900 Min. : 2.00 Min. : 9.0   
## 1st Qu.:0.03600 1st Qu.: 23.00 1st Qu.:108.0   
## Median :0.04300 Median : 34.00 Median :134.0   
## Mean :0.04577 Mean : 35.31 Mean :138.4   
## 3rd Qu.:0.05000 3rd Qu.: 46.00 3rd Qu.:167.0   
## Max. :0.34600 Max. :289.00 Max. :440.0   
## density pH sulphates alcohol   
## Min. :0.9871 Min. :2.720 Min. :0.2200 Min. : 8.00   
## 1st Qu.:0.9917 1st Qu.:3.090 1st Qu.:0.4100 1st Qu.: 9.50   
## Median :0.9937 Median :3.180 Median :0.4700 Median :10.40   
## Mean :0.9940 Mean :3.188 Mean :0.4898 Mean :10.51   
## 3rd Qu.:0.9961 3rd Qu.:3.280 3rd Qu.:0.5500 3rd Qu.:11.40   
## Max. :1.0390 Max. :3.820 Max. :1.0800 Max. :14.20   
## quality group   
## Min. :3.000 Min. :0.0002833   
## 1st Qu.:5.000 1st Qu.:0.2537537   
## Median :6.000 Median :0.5034640   
## Mean :5.878 Mean :0.5033795   
## 3rd Qu.:6.000 3rd Qu.:0.7564856   
## Max. :9.000 Max. :0.9993326

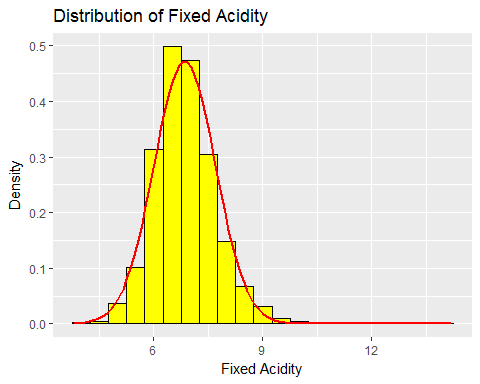
ww.train <- subset(ww, group <= 0.9)  
ww.test <- subset(ww, group > 0.9)

# Univariate Plot for Quality

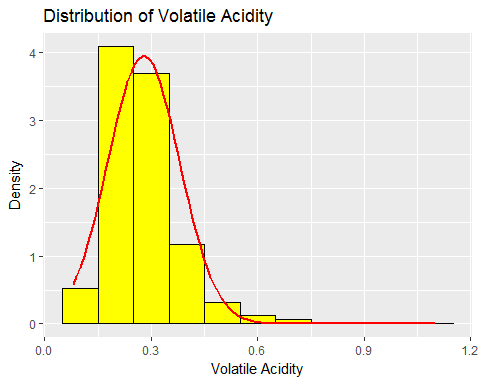
library(ggplot2)   
ww.train %>%  
 ggplot(aes(quality)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=1) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$quality),   
 sd = sd(ww.train$quality)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Quality",  
 x = "Quality",  
 y = "Density")

 #Univariate Plot for Fixed Acidity

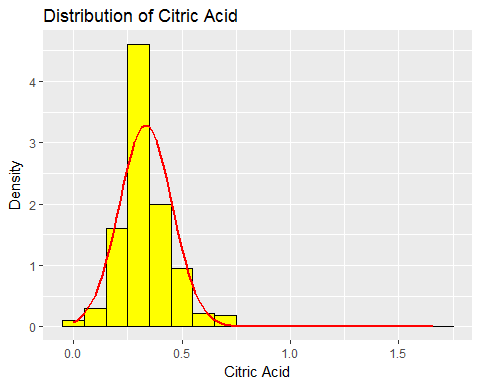
ww.train %>%  
 ggplot(aes(fixed.acidity)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=0.5) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$fixed.acidity),   
 sd = sd(ww.train$fixed.acidity)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Fixed Acidity",  
 x = "Fixed Acidity",  
 y = "Density")

 #Univariate Plot for Volatile Acidity

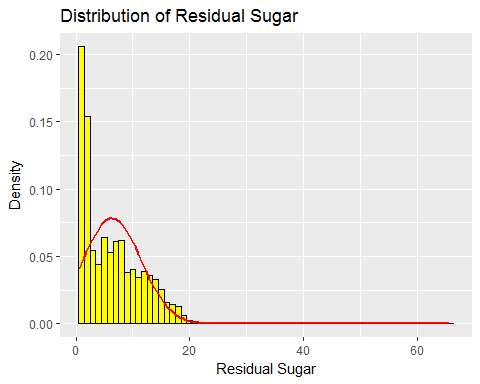
ww.train %>%  
 ggplot(aes(volatile.acidity)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=0.1) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$volatile.acidity),   
 sd = sd(ww.train$volatile.acidity)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Volatile Acidity",  
 x = "Volatile Acidity",  
 y = "Density")

 #Univariate Plot for Citric Acid

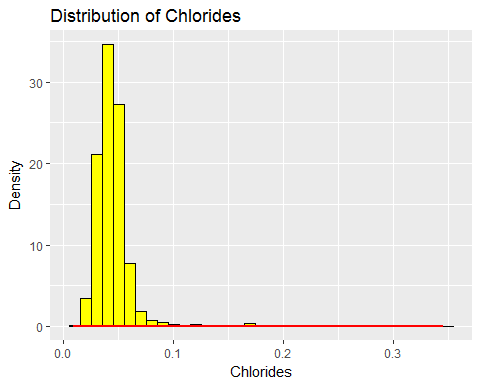
ww.train %>%  
 ggplot(aes(citric.acid)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=0.1) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$citric.acid),   
 sd = sd(ww.train$citric.acid)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Citric Acid",  
 x = "Citric Acid",  
 y = "Density")

 #Univariate Plot for Residual Sugar

ww.train %>%  
 ggplot(aes(residual.sugar)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=1) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$residual.sugar),   
 sd = sd(ww.train$residual.sugar)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Residual Sugar",  
 x = "Residual Sugar",  
 y = "Density")

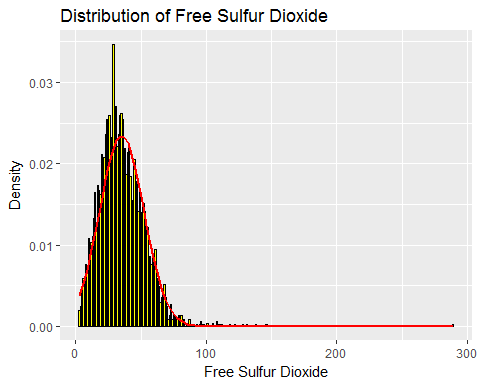
 #graphing Univariate Plot for Chlorides

ww.train %>%  
 ggplot(aes(chlorides)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=0.01) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$residual.sugar),   
 sd = sd(ww.train$residual.sugar)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Chlorides",  
 x = "Chlorides",  
 y = "Density")

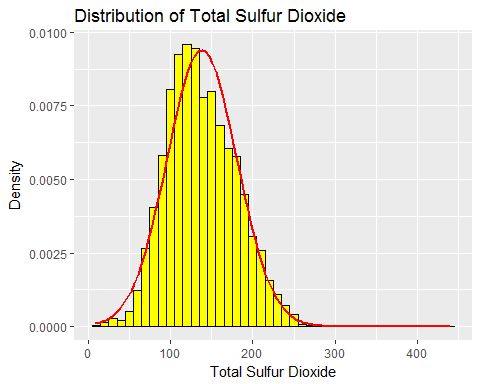


# Univariate Plot for Free Sulfur Dioxide

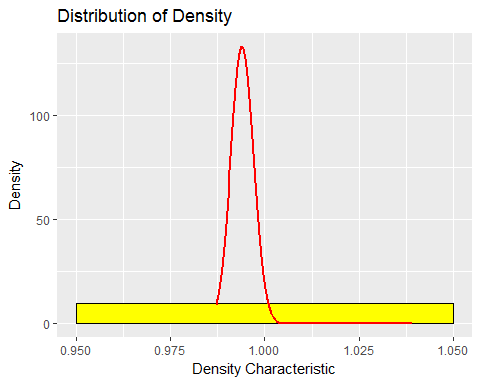
ww.train %>%  
 ggplot(aes(free.sulfur.dioxide)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=1) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$free.sulfur.dioxide),   
 sd = sd(ww.train$free.sulfur.dioxide)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Free Sulfur Dioxide",  
 x = "Free Sulfur Dioxide",  
 y = "Density")

 #Univariate Plot for Total Sulfur Dioxide

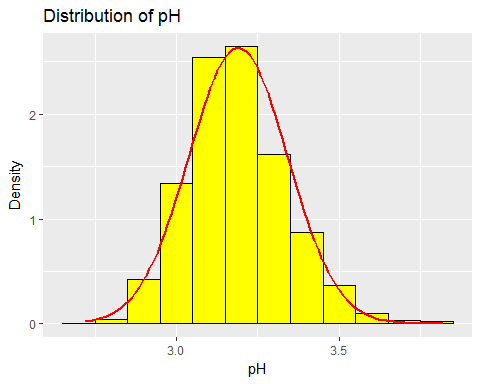
ww.train %>%  
 ggplot(aes(total.sulfur.dioxide)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=10) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$total.sulfur.dioxide),   
 sd = sd(ww.train$total.sulfur.dioxide)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Total Sulfur Dioxide",  
 x = "Total Sulfur Dioxide",  
 y = "Density")

 #graphing Univariate Plot for Density

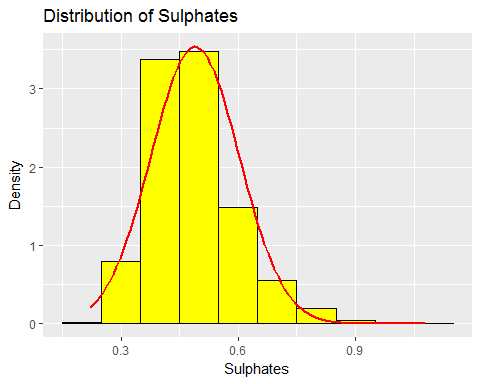
ww.train %>%  
 ggplot(aes(density)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=0.1) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$density),   
 sd = sd(ww.train$density)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Density",  
 x = "Density Characteristic",  
 y = "Density")

 #Univariate Plot for pH Sugar

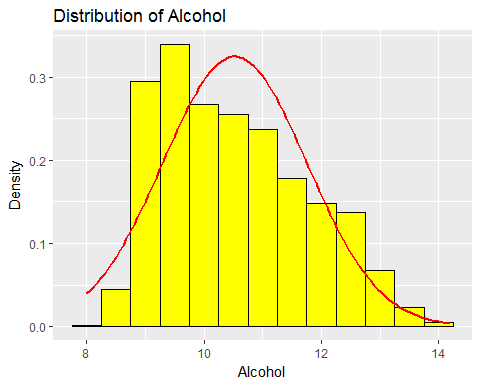
ww.train %>%  
 ggplot(aes(pH)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=0.1) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$pH),   
 sd = sd(ww.train$pH)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of pH",  
 x = "pH",  
 y = "Density")

 #Univariate Plot for Sulphates

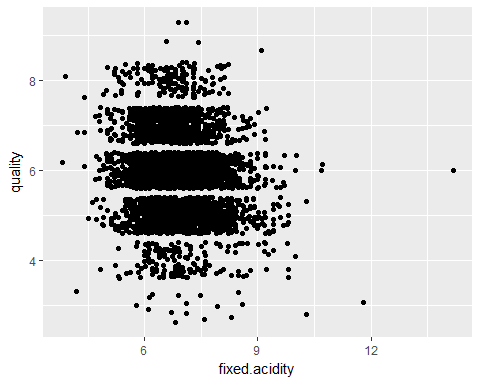
ww.train %>%  
 ggplot(aes(sulphates)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=0.1) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$sulphates),   
 sd = sd(ww.train$sulphates)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Sulphates",  
 x = "Sulphates",  
 y = "Density")

 #Univariate Plot for Alcohol

ww.train %>%  
 ggplot(aes(alcohol)) +  
 geom\_histogram(aes(y=..density..),  
 colour="black",fill="yellow",  
 binwidth=0.5) +   
 stat\_function(fun = dnorm,   
 args = list(mean = mean(ww.train$alcohol),   
 sd = sd(ww.train$alcohol)),   
 lwd = 1,   
 col = 'red') +  
 labs(title = "Distribution of Alcohol",  
 x = "Alcohol",  
 y = "Density")

 #Free Sulfer Dioxide against quality for training dataset

ww.train %>%  
 ggplot(aes(x=fixed.acidity, y=quality)) +  
 geom\_point(position = "jitter")

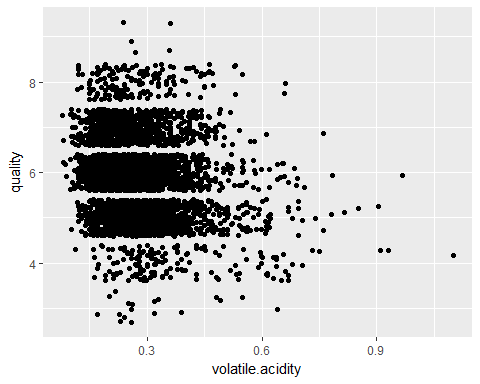


# linear regression of quality on fixed acidity in the training dataset  
reg1 <- lm(quality ~ fixed.acidity, data = ww.train)  
summary(reg1)

##   
## Call:  
## lm(formula = quality ~ fixed.acidity, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.1619 -0.8199 0.0839 0.2549 3.3618   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.61086 0.10815 61.128 < 2e-16 \*\*\*  
## fixed.acidity -0.10689 0.01567 -6.823 1.01e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8831 on 4436 degrees of freedom  
## Multiple R-squared: 0.01038, Adjusted R-squared: 0.01016   
## F-statistic: 46.55 on 1 and 4436 DF, p-value: 1.013e-11

# Scatterplot for volatile acidity against quality for training dataset

ww.train %>%  
 ggplot(aes(x=volatile.acidity, y=quality)) +  
 geom\_point(position = "jitter")

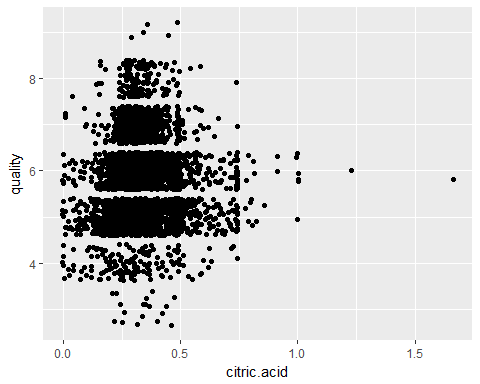


# linear regression of quality on volatile acidity in the training dataset  
reg2 <- lm(quality ~ volatile.acidity, data = ww.train)  
summary(reg2)

##   
## Call:  
## lm(formula = quality ~ volatile.acidity, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.0599 -0.7839 0.0238 0.3081 3.2579   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.34420 0.03832 165.55 <2e-16 \*\*\*  
## volatile.acidity -1.67256 0.12938 -12.93 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8714 on 4436 degrees of freedom  
## Multiple R-squared: 0.03631, Adjusted R-squared: 0.03609   
## F-statistic: 167.1 on 1 and 4436 DF, p-value: < 2.2e-16

# Scatterplot for citric acid against quality for training dataset

ww.train %>%  
 ggplot(aes(x=citric.acid, y=quality)) +  
 geom\_point(position = "jitter")

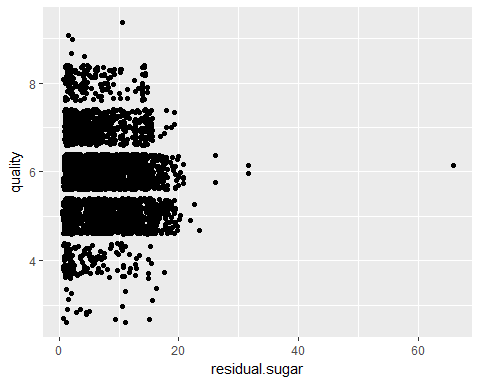


# linear regression of quality on citric acid in the training dataset  
reg4 <- lm(quality ~ citric.acid, data = ww.train)  
summary(reg4)

##   
## Call:  
## lm(formula = quality ~ citric.acid, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.8806 -0.8773 0.1214 0.1233 3.1221   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.87345 0.03902 150.517 <2e-16 \*\*\*  
## citric.acid 0.01527 0.10990 0.139 0.889   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8877 on 4436 degrees of freedom  
## Multiple R-squared: 4.353e-06, Adjusted R-squared: -0.0002211   
## F-statistic: 0.01931 on 1 and 4436 DF, p-value: 0.8895

# Scatterplot for residual sugar against quality for training dataset

ww.train %>%  
 ggplot(aes(x=residual.sugar, y=quality)) +  
 geom\_point(position = "jitter")

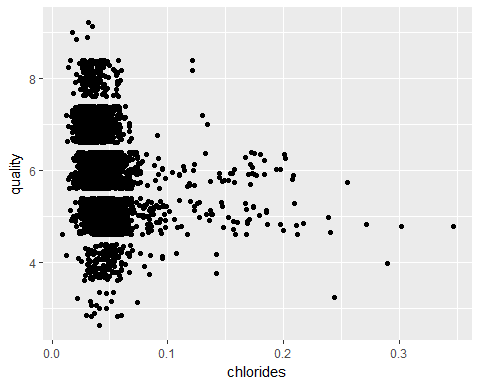


# linear regression of quality on residual sugar in the training dataset  
reg5 <- lm(quality ~ residual.sugar, data = ww.train)  
summary(reg5)

##   
## Call:  
## lm(formula = quality ~ residual.sugar, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.9666 -0.8137 0.0654 0.2487 3.1879   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.977488 0.021233 281.52 < 2e-16 \*\*\*  
## residual.sugar -0.015600 0.002613 -5.97 2.56e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8841 on 4436 degrees of freedom  
## Multiple R-squared: 0.007969, Adjusted R-squared: 0.007746   
## F-statistic: 35.64 on 1 and 4436 DF, p-value: 2.564e-09

# Scatterplot for chlorides against quality for training dataset

ww.train %>%  
 ggplot(aes(x=chlorides, y=quality)) +  
 geom\_point(position = "jitter")

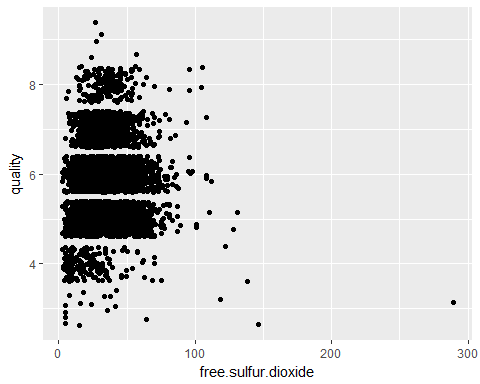


# linear regression of quality on chlorides in the training dataset  
reg6 <- lm(quality ~ chlorides, data = ww.train)  
summary(reg6)

##   
## Call:  
## lm(formula = quality ~ chlorides, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.07861 -0.82576 0.06467 0.24166 3.03096   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.2640 0.0300 208.80 <2e-16 \*\*\*  
## chlorides -8.4282 0.5908 -14.27 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.868 on 4436 degrees of freedom  
## Multiple R-squared: 0.04386, Adjusted R-squared: 0.04364   
## F-statistic: 203.5 on 1 and 4436 DF, p-value: < 2.2e-16

# Scatterplot for free sulfer dioxide against quality for training dataset

ww.train %>%  
 ggplot(aes(x=free.sulfur.dioxide, y=quality)) +  
 geom\_point(position = "jitter")

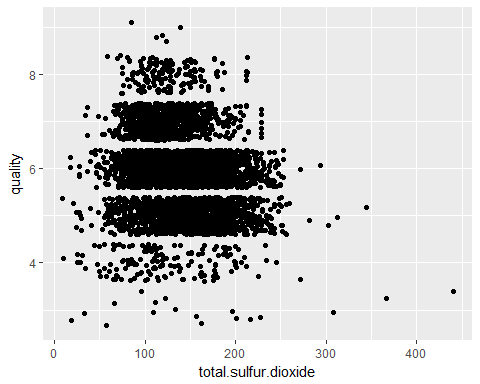


# linear regression of quality on free sulfur dioxide in the training dataset  
reg7 <- lm(quality ~ free.sulfur.dioxide, data = ww.train)  
summary(reg7)

##   
## Call:  
## lm(formula = quality ~ free.sulfur.dioxide, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.0746 -0.8698 0.1186 0.1371 3.1302   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.8512861 0.0306710 190.776 <2e-16 \*\*\*  
## free.sulfur.dioxide 0.0007728 0.0007831 0.987 0.324   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8876 on 4436 degrees of freedom  
## Multiple R-squared: 0.0002195, Adjusted R-squared: -5.886e-06   
## F-statistic: 0.9739 on 1 and 4436 DF, p-value: 0.3238

Scatterplot for total sulfer dioxide against quality for training dataset

ww.train %>%  
 ggplot(aes(x=total.sulfur.dioxide, y=quality)) +  
 geom\_point(position = "jitter")

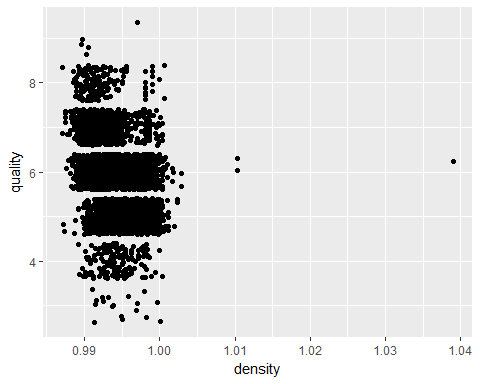


# linear regression of quality on total sulfur dioxide in the training dataset  
reg8 <- lm(quality ~ total.sulfur.dioxide, data = ww.train)  
summary(reg8)

##   
## Call:  
## lm(formula = quality ~ total.sulfur.dioxide, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.3044 -0.7393 0.0491 0.3455 3.1241   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.3722105 0.0446725 142.64 <2e-16 \*\*\*  
## total.sulfur.dioxide -0.0035707 0.0003089 -11.56 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8746 on 4436 degrees of freedom  
## Multiple R-squared: 0.02925, Adjusted R-squared: 0.02903   
## F-statistic: 133.7 on 1 and 4436 DF, p-value: < 2.2e-16

Scatterplot for density against quality for training dataset

ww.train %>%  
 ggplot(aes(x=density, y=quality)) +  
 geom\_point(position = "jitter")

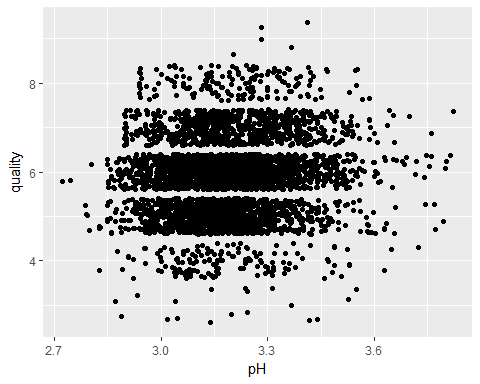


# linear regression of quality on density in the training dataset  
reg9 <- lm(quality ~ density, data = ww.train)  
summary(reg9)

##   
## Call:  
## lm(formula = quality ~ density, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.1353 -0.6317 0.0029 0.5095 4.0954   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 93.711 4.217 22.22 <2e-16 \*\*\*  
## density -88.362 4.243 -20.83 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8472 on 4436 degrees of freedom  
## Multiple R-squared: 0.08908, Adjusted R-squared: 0.08887   
## F-statistic: 433.8 on 1 and 4436 DF, p-value: < 2.2e-16

Scatterplot for pH against quality for training dataset

ww.train %>%  
 ggplot(aes(x=pH, y=quality)) +  
 geom\_point(position = "jitter")

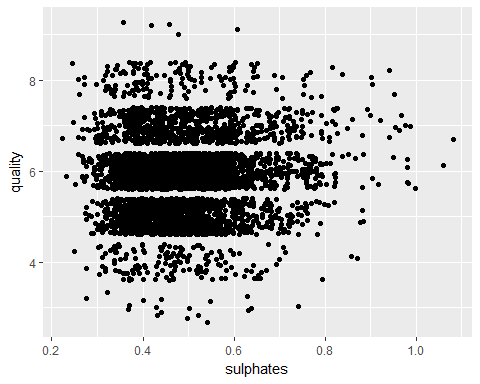


# linear regression of quality on pH in the training dataset  
reg10 <- lm(quality ~ pH, data = ww.train)  
summary(reg10)

##   
## Call:  
## lm(formula = quality ~ pH, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.0875 -0.8207 0.0981 0.2373 3.1155   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.02861 0.27965 14.406 < 2e-16 \*\*\*  
## pH 0.57996 0.08757 6.623 3.95e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8833 on 4436 degrees of freedom  
## Multiple R-squared: 0.009791, Adjusted R-squared: 0.009567   
## F-statistic: 43.86 on 1 and 4436 DF, p-value: 3.947e-11

Scatterplot for sulphates against quality for training dataset

ww.train %>%  
 ggplot(aes(x=sulphates, y=quality)) +  
 geom\_point(position = "jitter")

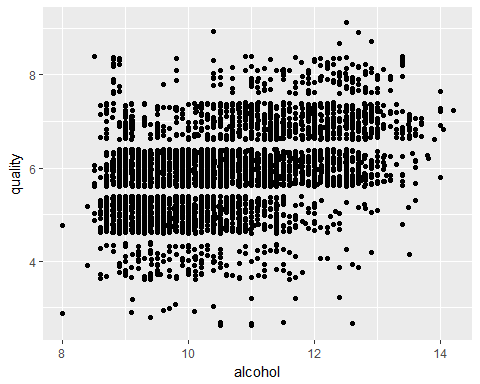


# linear regression of quality on sulfates in the training dataset  
reg11 <- lm(quality ~ sulphates, data = ww.train)  
summary(reg11)

##   
## Call:  
## lm(formula = quality ~ sulphates, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.9799 -0.8506 0.1130 0.1777 3.1737   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.68082 0.05922 95.931 < 2e-16 \*\*\*  
## sulphates 0.40418 0.11795 3.427 0.000617 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8865 on 4436 degrees of freedom  
## Multiple R-squared: 0.00264, Adjusted R-squared: 0.002415   
## F-statistic: 11.74 on 1 and 4436 DF, p-value: 0.0006167

Scatterplot for alcohol against quality for training dataset

ww.train %>%  
 ggplot(aes(x=alcohol, y=quality)) +  
 geom\_point(position = "jitter")

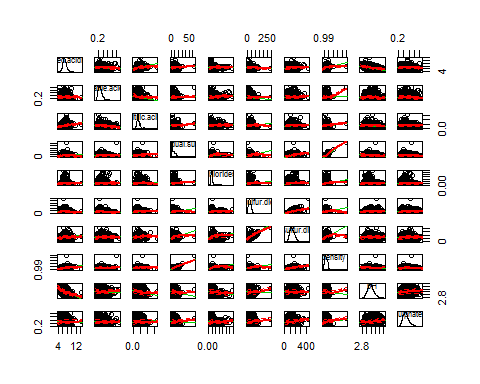


# linear regression of quality on alcohol in the training dataset  
reg12 <- lm(quality ~ alcohol, data = ww.train)  
summary(reg12)

##   
## Call:  
## lm(formula = quality ~ alcohol, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.5282 -0.5284 -0.0022 0.5029 3.1592   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.591364 0.103635 25.00 <2e-16 \*\*\*  
## alcohol 0.312446 0.009784 31.93 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8004 on 4436 degrees of freedom  
## Multiple R-squared: 0.1869, Adjusted R-squared: 0.1867   
## F-statistic: 1020 on 1 and 4436 DF, p-value: < 2.2e-16

Creating a scatterplot matrix for the predictor variables

#produce a scatterplot matrix   
scatterplotMatrix(~ fixed.acidity + volatile.acidity + citric.acid + residual.sugar + chlorides + free.sulfur.dioxide + total.sulfur.dioxide + density + pH + sulphates, span =0.7, data = ww.train)

 Look at correlations amongst the variables

cor(ww.train, method = c("pearson"))

## fixed.acidity volatile.acidity citric.acid  
## fixed.acidity 1.000000000 -0.02889990 0.292972820  
## volatile.acidity -0.028899905 1.00000000 -0.157831967  
## citric.acid 0.292972820 -0.15783197 1.000000000  
## residual.sugar 0.089398508 0.06624507 0.089964462  
## chlorides 0.027629785 0.07388087 0.109817383  
## free.sulfur.dioxide -0.050145093 -0.10015954 0.101863843  
## total.sulfur.dioxide 0.099158024 0.08447918 0.126180725  
## density 0.265784160 0.02500269 0.143825322  
## pH -0.423043715 -0.03398943 -0.162003175  
## sulphates -0.014914324 -0.03987275 0.065167623  
## alcohol -0.118126116 0.07186987 -0.066282341  
## quality -0.101906285 -0.19054240 0.002086427  
## group -0.006874152 -0.03314154 -0.016869732  
## residual.sugar chlorides free.sulfur.dioxide  
## fixed.acidity 0.08939851 0.027629785 -0.050145093  
## volatile.acidity 0.06624507 0.073880870 -0.100159545  
## citric.acid 0.08996446 0.109817383 0.101863843  
## residual.sugar 1.00000000 0.090300949 0.301936927  
## chlorides 0.09030095 1.000000000 0.104534178  
## free.sulfur.dioxide 0.30193693 0.104534178 1.000000000  
## total.sulfur.dioxide 0.40861589 0.203893380 0.613426897  
## density 0.83916403 0.257313339 0.294344481  
## pH -0.19371439 -0.094971625 -0.001305664  
## sulphates -0.02410850 0.024848259 0.065680255  
## alcohol -0.44595149 -0.358929650 -0.246801036  
## quality -0.08927177 -0.209422969 0.014815277  
## group 0.00689372 0.001741242 0.014553846  
## total.sulfur.dioxide density pH  
## fixed.acidity 0.099158024 0.26578416 -0.423043715  
## volatile.acidity 0.084479180 0.02500269 -0.033989432  
## citric.acid 0.126180725 0.14382532 -0.162003175  
## residual.sugar 0.408615891 0.83916403 -0.193714393  
## chlorides 0.203893380 0.25731334 -0.094971625  
## free.sulfur.dioxide 0.613426897 0.29434448 -0.001305664  
## total.sulfur.dioxide 1.000000000 0.53152450 -0.008767048  
## density 0.531524504 1.00000000 -0.094439253  
## pH -0.008767048 -0.09443925 1.000000000  
## sulphates 0.134543538 0.08102247 0.159660669  
## alcohol -0.447451766 -0.77541077 0.122837332  
## quality -0.171026003 -0.29845837 0.098947136  
## group -0.012095898 0.01719093 -0.003587772  
## sulphates alcohol quality group  
## fixed.acidity -0.01491432 -0.11812612 -0.101906285 -0.006874152  
## volatile.acidity -0.03987275 0.07186987 -0.190542402 -0.033141542  
## citric.acid 0.06516762 -0.06628234 0.002086427 -0.016869732  
## residual.sugar -0.02410850 -0.44595149 -0.089271768 0.006893720  
## chlorides 0.02484826 -0.35892965 -0.209422969 0.001741242  
## free.sulfur.dioxide 0.06568026 -0.24680104 0.014815277 0.014553846  
## total.sulfur.dioxide 0.13454354 -0.44745177 -0.171026003 -0.012095898  
## density 0.08102247 -0.77541077 -0.298458366 0.017190931  
## pH 0.15966067 0.12283733 0.098947136 -0.003587772  
## sulphates 1.00000000 -0.02826118 0.051380078 -0.017321185  
## alcohol -0.02826118 1.00000000 0.432343105 -0.030578479  
## quality 0.05138008 0.43234311 1.000000000 0.001873573  
## group -0.01732118 -0.03057848 0.001873573 1.000000000

# make a correlation matrix with significance values

library("Hmisc")

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

##   
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':  
##   
## combine, src, summarize

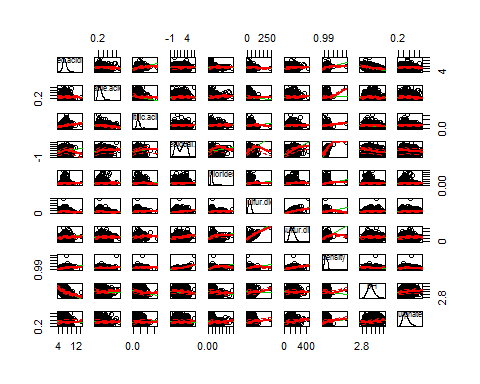
## The following objects are masked from 'package:base':  
##   
## format.pval, round.POSIXt, trunc.POSIXt, units

res2 <- rcorr(as.matrix(ww.train))  
res2

## fixed.acidity volatile.acidity citric.acid  
## fixed.acidity 1.00 -0.03 0.29  
## volatile.acidity -0.03 1.00 -0.16  
## citric.acid 0.29 -0.16 1.00  
## residual.sugar 0.09 0.07 0.09  
## chlorides 0.03 0.07 0.11  
## free.sulfur.dioxide -0.05 -0.10 0.10  
## total.sulfur.dioxide 0.10 0.08 0.13  
## density 0.27 0.03 0.14  
## pH -0.42 -0.03 -0.16  
## sulphates -0.01 -0.04 0.07  
## alcohol -0.12 0.07 -0.07  
## quality -0.10 -0.19 0.00  
## group -0.01 -0.03 -0.02  
## residual.sugar chlorides free.sulfur.dioxide  
## fixed.acidity 0.09 0.03 -0.05  
## volatile.acidity 0.07 0.07 -0.10  
## citric.acid 0.09 0.11 0.10  
## residual.sugar 1.00 0.09 0.30  
## chlorides 0.09 1.00 0.10  
## free.sulfur.dioxide 0.30 0.10 1.00  
## total.sulfur.dioxide 0.41 0.20 0.61  
## density 0.84 0.26 0.29  
## pH -0.19 -0.09 0.00  
## sulphates -0.02 0.02 0.07  
## alcohol -0.45 -0.36 -0.25  
## quality -0.09 -0.21 0.01  
## group 0.01 0.00 0.01  
## total.sulfur.dioxide density pH sulphates alcohol  
## fixed.acidity 0.10 0.27 -0.42 -0.01 -0.12  
## volatile.acidity 0.08 0.03 -0.03 -0.04 0.07  
## citric.acid 0.13 0.14 -0.16 0.07 -0.07  
## residual.sugar 0.41 0.84 -0.19 -0.02 -0.45  
## chlorides 0.20 0.26 -0.09 0.02 -0.36  
## free.sulfur.dioxide 0.61 0.29 0.00 0.07 -0.25  
## total.sulfur.dioxide 1.00 0.53 -0.01 0.13 -0.45  
## density 0.53 1.00 -0.09 0.08 -0.78  
## pH -0.01 -0.09 1.00 0.16 0.12  
## sulphates 0.13 0.08 0.16 1.00 -0.03  
## alcohol -0.45 -0.78 0.12 -0.03 1.00  
## quality -0.17 -0.30 0.10 0.05 0.43  
## group -0.01 0.02 0.00 -0.02 -0.03  
## quality group  
## fixed.acidity -0.10 -0.01  
## volatile.acidity -0.19 -0.03  
## citric.acid 0.00 -0.02  
## residual.sugar -0.09 0.01  
## chlorides -0.21 0.00  
## free.sulfur.dioxide 0.01 0.01  
## total.sulfur.dioxide -0.17 -0.01  
## density -0.30 0.02  
## pH 0.10 0.00  
## sulphates 0.05 -0.02  
## alcohol 0.43 -0.03  
## quality 1.00 0.00  
## group 0.00 1.00  
##   
## n= 4438   
##   
##   
## P  
## fixed.acidity volatile.acidity citric.acid  
## fixed.acidity 0.0542 0.0000   
## volatile.acidity 0.0542 0.0000   
## citric.acid 0.0000 0.0000   
## residual.sugar 0.0000 0.0000 0.0000   
## chlorides 0.0657 0.0000 0.0000   
## free.sulfur.dioxide 0.0008 0.0000 0.0000   
## total.sulfur.dioxide 0.0000 0.0000 0.0000   
## density 0.0000 0.0958 0.0000   
## pH 0.0000 0.0236 0.0000   
## sulphates 0.3205 0.0079 0.0000   
## alcohol 0.0000 0.0000 0.0000   
## quality 0.0000 0.0000 0.8895   
## group 0.6471 0.0273 0.2612   
## residual.sugar chlorides free.sulfur.dioxide  
## fixed.acidity 0.0000 0.0657 0.0008   
## volatile.acidity 0.0000 0.0000 0.0000   
## citric.acid 0.0000 0.0000 0.0000   
## residual.sugar 0.0000 0.0000   
## chlorides 0.0000 0.0000   
## free.sulfur.dioxide 0.0000 0.0000   
## total.sulfur.dioxide 0.0000 0.0000 0.0000   
## density 0.0000 0.0000 0.0000   
## pH 0.0000 0.0000 0.9307   
## sulphates 0.1083 0.0979 0.0000   
## alcohol 0.0000 0.0000 0.0000   
## quality 0.0000 0.0000 0.3238   
## group 0.6461 0.9077 0.3324   
## total.sulfur.dioxide density pH sulphates alcohol  
## fixed.acidity 0.0000 0.0000 0.0000 0.3205 0.0000   
## volatile.acidity 0.0000 0.0958 0.0236 0.0079 0.0000   
## citric.acid 0.0000 0.0000 0.0000 0.0000 0.0000   
## residual.sugar 0.0000 0.0000 0.0000 0.1083 0.0000   
## chlorides 0.0000 0.0000 0.0000 0.0979 0.0000   
## free.sulfur.dioxide 0.0000 0.0000 0.9307 0.0000 0.0000   
## total.sulfur.dioxide 0.0000 0.5593 0.0000 0.0000   
## density 0.0000 0.0000 0.0000 0.0000   
## pH 0.5593 0.0000 0.0000 0.0000   
## sulphates 0.0000 0.0000 0.0000 0.0598   
## alcohol 0.0000 0.0000 0.0000 0.0598   
## quality 0.0000 0.0000 0.0000 0.0006 0.0000   
## group 0.4205 0.2522 0.8111 0.2486 0.0417   
## quality group   
## fixed.acidity 0.0000 0.6471  
## volatile.acidity 0.0000 0.0273  
## citric.acid 0.8895 0.2612  
## residual.sugar 0.0000 0.6461  
## chlorides 0.0000 0.9077  
## free.sulfur.dioxide 0.3238 0.3324  
## total.sulfur.dioxide 0.0000 0.4205  
## density 0.0000 0.2522  
## pH 0.0000 0.8111  
## sulphates 0.0006 0.2486  
## alcohol 0.0000 0.0417  
## quality 0.9007  
## group 0.9007

Creating a scatterplot matrix with log2 transformation of residual sugar due to "wonky" plot in scatterplot matrix

#create a scatterplot matrix   
scatterplotMatrix(~ fixed.acidity + volatile.acidity + citric.acid + log2(residual.sugar)+1 + chlorides + free.sulfur.dioxide + total.sulfur.dioxide + density + pH + sulphates, span =0.7, data = ww.train)

 Creating a model, looking at the model of the regression for quality on fixed acidity, volatile acidity, residual sugar(logged), chlorides, total sulfur dioxide, density, pH, sulphates, alcohol (This is the full model with all of the potential predictor variables excet for citric acid and free sulfur dioxide which were not statistically significant in the simple linear regressions)

# perform the regression  
ww.mod1 <- lm(quality ~ fixed.acidity + volatile.acidity + log2(residual.sugar) + chlorides + total.sulfur.dioxide + density + pH + sulphates + alcohol, data= ww.train)  
  
#summary of the regression  
summary(ww.mod1)

##   
## Call:  
## lm(formula = quality ~ fixed.acidity + volatile.acidity + log2(residual.sugar) +   
## chlorides + total.sulfur.dioxide + density + pH + sulphates +   
## alcohol, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.4884 -0.4878 -0.0512 0.4669 3.0900   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.369e+01 1.176e+01 4.567 5.08e-06 \*\*\*  
## fixed.acidity -7.952e-03 1.789e-02 -0.444 0.656774   
## volatile.acidity -2.065e+00 1.157e-01 -17.848 < 2e-16 \*\*\*  
## log2(residual.sugar) 1.788e-01 1.767e-02 10.122 < 2e-16 \*\*\*  
## chlorides -7.684e-01 5.630e-01 -1.365 0.172380   
## total.sulfur.dioxide 1.649e-04 3.238e-04 0.509 0.610515   
## density -5.227e+01 1.189e+01 -4.396 1.13e-05 \*\*\*  
## pH 3.299e-01 9.410e-02 3.506 0.000459 \*\*\*  
## sulphates 5.210e-01 1.052e-01 4.952 7.60e-07 \*\*\*  
## alcohol 2.944e-01 1.936e-02 15.211 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7585 on 4428 degrees of freedom  
## Multiple R-squared: 0.2712, Adjusted R-squared: 0.2698   
## F-statistic: 183.1 on 9 and 4428 DF, p-value: < 2.2e-16

Now I am going to look at the model of the regression for quality on volatile acidity, citric acid, residual sugar(logged), free sulfur dioxide,pH, sulphates, alcohol (fixed acidity, chlorides and total sulfur dioxide removed becauise they were not significant in model 1)

# performing a regression for this more parsimonious model with out fixed acidity, chlorides or total sulfur dioxide  
ww.mod2 <- lm(quality ~ log2(residual.sugar) + density + pH + sulphates + alcohol, data= ww.train)  
  
#summary of the regression  
summary(ww.mod2)

##   
## Call:  
## lm(formula = quality ~ log2(residual.sugar) + density + pH +   
## sulphates + alcohol, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.5713 -0.5371 -0.0137 0.4790 3.1342   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 65.45408 10.27764 6.369 2.10e-10 \*\*\*  
## log2(residual.sugar) 0.17532 0.01590 11.026 < 2e-16 \*\*\*  
## density -64.64215 10.26111 -6.300 3.27e-10 \*\*\*  
## pH 0.40276 0.08130 4.954 7.53e-07 \*\*\*  
## sulphates 0.59937 0.10789 5.556 2.93e-08 \*\*\*  
## alcohol 0.25945 0.01768 14.679 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7861 on 4432 degrees of freedom  
## Multiple R-squared: 0.2166, Adjusted R-squared: 0.2157   
## F-statistic: 245.1 on 5 and 4432 DF, p-value: < 2.2e-16

Comparing Models

library(stargazer)  
# compare the results of three regression models - if you want it to look pretty do type=html when using stargazer  
stargazer(ww.mod1,ww.mod2, title="Comparison of 2 Regression outputs",type="text",align=TRUE)

# Comparison of 2 Regression outputs

Dependent variable:   
 ---------------------------------------------------  
 quality   
 (1) (2)

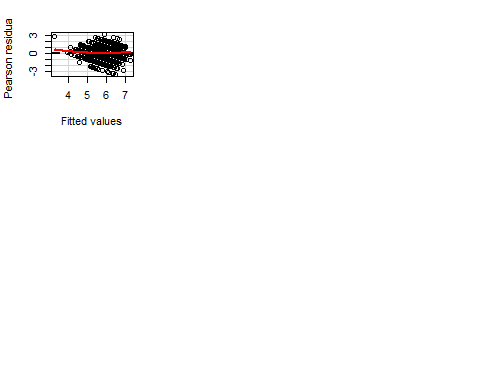
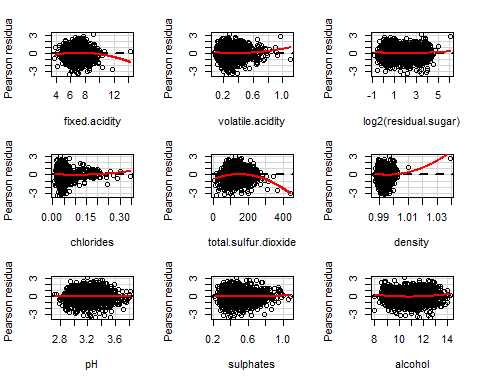
|  |
| --- |
| fixed.acidity -0.008 (0.018) |
| volatile.acidity -2.065\*\*\* (0.116) |
| log2(residual.sugar) 0.179\*\*\* 0.175\*\*\* (0.018) (0.016) |
| chlorides -0.768 (0.563) |
| total.sulfur.dioxide 0.0002 (0.0003) |
| density -52.266\*\*\* -64.642\*\*\* (11.891) (10.261) |
| pH 0.330\*\*\* 0.403\*\*\* (0.094) (0.081) |
| sulphates 0.521\*\*\* 0.599\*\*\* (0.105) (0.108) |
| alcohol 0.294\*\*\* 0.259\*\*\* (0.019) (0.018) |
| Constant 53.689\*\*\* 65.454\*\*\* (11.756) (10.278) |

Observations 4,438 4,438  
R2 0.271 0.217  
Adjusted R2 0.270 0.216  
Residual Std. Error 0.758 (df = 4428) 0.786 (df = 4432)  
F Statistic 183.121\*\*\* (df = 9; 4428) 245.054\*\*\* (df = 5; 4432) ======================================================================== Note: *p<0.1;* ***p<0.05;*** p<0.01 When comparing mod1 to mod2, the multiple R2 decreased by .5, but mod2 was more parsimonious and had an larger F statistic than mod1.

Now to do diagnostics for the 2 models:

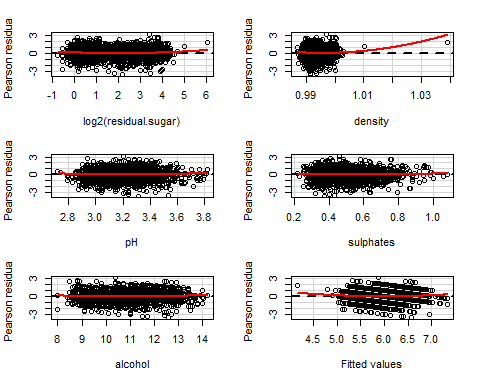
Looking at residual plots first

# diagnostics for the model 1  
residualPlots(ww.mod1)



## Test stat Pr(>|t|)  
## fixed.acidity -3.722 0.000  
## volatile.acidity 3.395 0.001  
## log2(residual.sugar) 3.436 0.001  
## chlorides 1.847 0.065  
## total.sulfur.dioxide -9.375 0.000  
## density 7.778 0.000  
## pH 0.850 0.395  
## sulphates 1.105 0.269  
## alcohol 6.190 0.000  
## Tukey test 2.711 0.007

# diagnostics for the model 2  
residualPlots(ww.mod2)

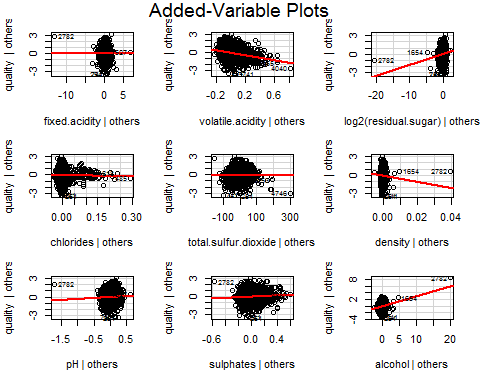


## Test stat Pr(>|t|)  
## log2(residual.sugar) 4.860 0.000  
## density 5.258 0.000  
## pH 2.366 0.018  
## sulphates 1.426 0.154  
## alcohol 2.440 0.015  
## Tukey test 3.193 0.001

The residual plots llook much better now, as compard to mod1, with the exception of the variable density, but it looks like this is due to a few potential outliers.

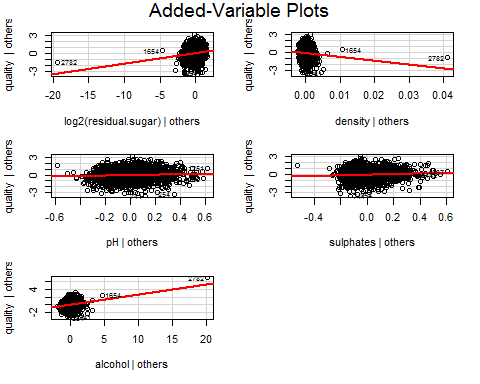
Next I looked at the added variable plots, which shows the additional benefit of variable *i* given that all of the others are in. In this particular plot the most influential observations can also be identified.

#added variable plots  
avPlots(ww.mod1, id.n=2, id.cex=0.7)



#id.n - identify n most influential observations  
#id.cex - controls the size of the dot

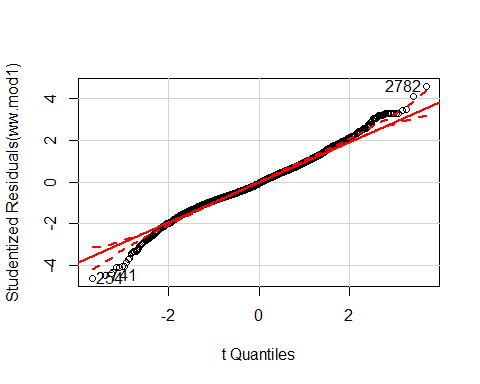
#added variable plots  
avPlots(ww.mod2, id.n=2, id.cex=0.7)



#id.n - identify n most influential observations  
#id.cex - controls the size of the dot

Next I ran the qq-plots

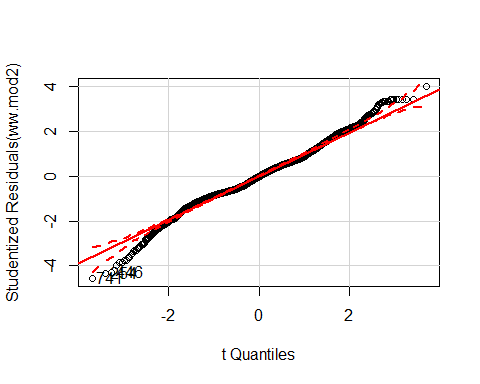
# run the qq-plot  
qqPlot(ww.mod1, id.n=3)



## 254 741 2782   
## 1 2 4438

# here, id.n identifies the n observations with the largest residuals in absolute value

# run the qq-plot  
qqPlot(ww.mod2, id.n=3)



## 741 254 446   
## 1 2 3

# here, id.n identifies the n observations with the largest residuals in absolute value

Checking for Outliers

#run Bonferroni test for outliers  
outlierTest(ww.mod1)

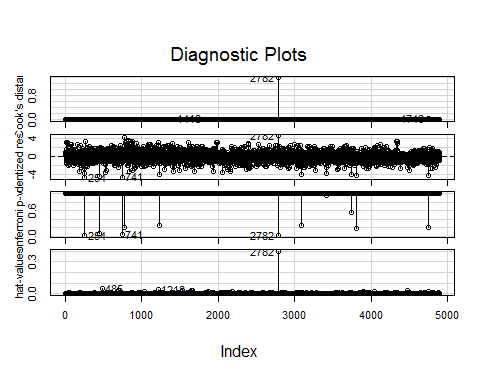
## rstudent unadjusted p-value Bonferonni p  
## 254 -4.614130 4.0590e-06 0.018014  
## 2782 4.590540 4.5435e-06 0.020164  
## 741 -4.475661 7.8089e-06 0.034656

#run Bonferroni test for outliers  
outlierTest(ww.mod1)

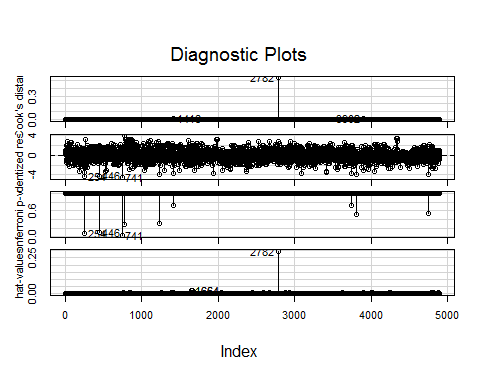
## rstudent unadjusted p-value Bonferonni p  
## 254 -4.614130 4.0590e-06 0.018014  
## 2782 4.590540 4.5435e-06 0.020164  
## 741 -4.475661 7.8089e-06 0.034656

Looking for points of high influence

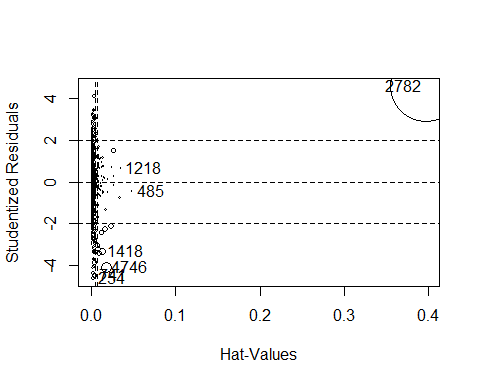
#identify highly influential points  
influenceIndexPlot(ww.mod1, id.n=3)



#identify highly influential points  
influenceIndexPlot(ww.mod2, id.n=3)

 Now for influence plots.

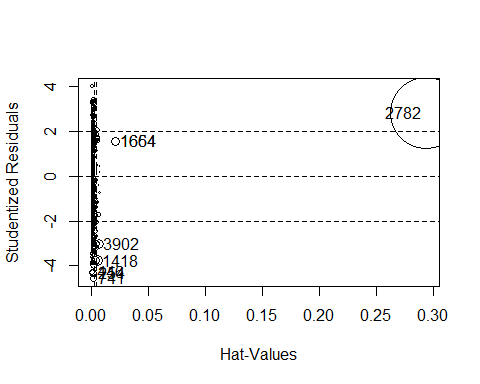
#make influence plot  
influencePlot(ww.mod1, id.n=3)



## StudRes Hat CookD  
## 254 -4.6141296 0.001927744 0.0040933748  
## 485 -0.4299817 0.048171285 0.0009358564  
## 741 -4.4756608 0.003436084 0.0068771791  
## 1218 0.6529779 0.034795988 0.0015373164  
## 1418 -3.3314148 0.012447512 0.0139569490  
## 2782 4.5905397 0.397461952 1.3838030276  
## 4746 -4.0867416 0.017404045 0.0294776134

How do we make heads or tails out of the plots above? One way is with an influence plot.

#make influence plot  
influencePlot(ww.mod2, id.n=3)



## StudRes Hat CookD  
## 254 -4.345969 0.001845407 0.005796517  
## 446 -4.277410 0.001049553 0.003191386  
## 741 -4.556644 0.001417723 0.004891185  
## 1418 -3.774567 0.004848030 0.011533543  
## 1654 1.544220 0.020882850 0.008473959  
## 1664 1.544220 0.020882850 0.008473959  
## 2782 2.778830 0.293597964 0.534091252  
## 3902 -3.039007 0.006464192 0.009996236

Because data entry 2782 was found to be an outlier and highly influential, it should be removed.

Next I performed the test for heteroskedasticity (i.e., the variance of the error term is not constant).

#test for heteroskedasticity  
ncvTest(ww.mod1)

## Non-constant Variance Score Test   
## Variance formula: ~ fitted.values   
## Chisquare = 10.28538 Df = 1 p = 0.001340882

#test for heteroskedasticity  
ncvTest(ww.mod2)

## Non-constant Variance Score Test   
## Variance formula: ~ fitted.values   
## Chisquare = 0.7036879 Df = 1 p = 0.4015473

mod2 remains superior to mod1, especially after a non-significant test for heteroskidasticity as compared to mod 1.

Next I looked for multicollinearity to see if some of the independent variables are highly correlated by looking at the Variance Inflation Factor (VIF).

vif(ww.mod1)

## fixed.acidity volatile.acidity log2(residual.sugar)   
## 1.768329 1.055457 4.279630   
## chlorides total.sulfur.dioxide density   
## 1.189262 1.461739 9.800981   
## pH sulphates alcohol   
## 1.566107 1.086842 4.358746

vif(ww.mod2)

## log2(residual.sugar) density pH   
## 3.228160 6.795683 1.088290   
## sulphates alcohol   
## 1.064075 3.384124

A GVIF > 4 suggests collinearity. In mod2, which, density has a VIFof 6.795, and will be removed from the model because it is highly correlated with another variable.

The final model that I am proposing is mod2, but without density, but I need to run mod2 again with and without density to compare.

I should also consider running these models without observation 2782 because it was found to be an outlier and highly influential, but I have run out of time to look that procedure up and perform it for this assignment.

# performing a regression (mod3) without density due to multicollinearity with other variables and the lack of variability in density  
ww.mod3 <- lm(quality ~ log2(residual.sugar) + pH + sulphates + alcohol, data= ww.train)  
  
#summary of the regression  
summary(ww.mod3)

##   
## Call:  
## lm(formula = quality ~ log2(residual.sugar) + pH + sulphates +   
## alcohol, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.5948 -0.5319 -0.0075 0.4763 3.0508   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.731326 0.278344 2.627 0.00863 \*\*   
## log2(residual.sugar) 0.095960 0.009745 9.847 < 2e-16 \*\*\*  
## pH 0.325076 0.080705 4.028 5.72e-05 \*\*\*  
## sulphates 0.476166 0.106560 4.469 8.07e-06 \*\*\*  
## alcohol 0.349206 0.010505 33.241 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7895 on 4433 degrees of freedom  
## Multiple R-squared: 0.2096, Adjusted R-squared: 0.2089   
## F-statistic: 293.8 on 4 and 4433 DF, p-value: < 2.2e-16

Comparing mod2 to mod3-which is the model without density

library(stargazer)  
# compare the results of three regression models - if you want it to look pretty do type=html when using stargazer  
stargazer(ww.mod2,ww.mod3, title="Comparison of 2 Regression outputs",type="text",align=TRUE)

# Comparison of 2 Regression outputs

Dependent variable:   
 ---------------------------------------------------  
 quality   
 (1) (2)

|  |
| --- |
| log2(residual.sugar) 0.175\*\*\* 0.096\*\*\* (0.016) (0.010) |
| density -64.642\*\*\* (10.261) |
| pH 0.403\*\*\* 0.325\*\*\* (0.081) (0.081) |
| sulphates 0.599\*\*\* 0.476\*\*\* (0.108) (0.107) |
| alcohol 0.259\*\*\* 0.349\*\*\* (0.018) (0.011) |
| Constant 65.454\*\*\* 0.731\*\*\* (10.278) (0.278) |

Observations 4,438 4,438  
R2 0.217 0.210  
Adjusted R2 0.216 0.209  
Residual Std. Error 0.786 (df = 4432) 0.789 (df = 4433)  
F Statistic 245.054\*\*\* (df = 5; 4432) 293.832\*\*\* (df = 4; 4433) ======================================================================== Note: *p<0.1;* ***p<0.05;*** p<0.01

# performing a regression for this more parsimonious model with out fixed acidity, chlorides or total sulfur dioxide  
ww.mod3 <- lm(quality ~ log2(residual.sugar) + pH + sulphates + alcohol, data= ww.train)  
  
#summary of the regression  
summary(ww.mod3)

##   
## Call:  
## lm(formula = quality ~ log2(residual.sugar) + pH + sulphates +   
## alcohol, data = ww.train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.5948 -0.5319 -0.0075 0.4763 3.0508   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.731326 0.278344 2.627 0.00863 \*\*   
## log2(residual.sugar) 0.095960 0.009745 9.847 < 2e-16 \*\*\*  
## pH 0.325076 0.080705 4.028 5.72e-05 \*\*\*  
## sulphates 0.476166 0.106560 4.469 8.07e-06 \*\*\*  
## alcohol 0.349206 0.010505 33.241 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7895 on 4433 degrees of freedom  
## Multiple R-squared: 0.2096, Adjusted R-squared: 0.2089   
## F-statistic: 293.8 on 4 and 4433 DF, p-value: < 2.2e-16

Based on these results, the final model I chose is mod3 because it has the higher F statistic, is more parsimonious and not much of the adjusted R2 was sacrificed when density was removed.

The final model is our estimated line is:

According to the output from this model (mod2), we see that if the log2 of residual sugar increases by 1 unit, then the wine quality rating will increase by 0.10 units, with pH + sulphates + alcohol held constant. If pH is increased by one unit, then the wine quality rating is expected to increase by 0.33 units, with all other variables held constant. Next, if the variable sulphates is increased by one unit, the wine quality rating is expected to increased by 0.48 units, holding all other variables constant. Lastly, for a one unit rise in alcohol, we expect wine quality to increase by 0.35 units.The multiple R-squared is 0.21, indicating that 22% of the variability in the wine quality rating is due to these three independent variables.

Finally, the F-statistic of 165.4 is for testing the null hypothesis H\_0: \_1 = \_2 = \_3 = \_4= 0. It is highly significant and we can therefore reject the H\_0.

Now I will test the chosen model (mod3) on the white wines test dataset I created (ww.test)

#performing regression on mod3  
ww.mod3test <- lm(quality ~ log2(residual.sugar) + pH + sulphates + alcohol, data= ww.test)  
  
#summary of the regression  
summary(ww.mod3test)

##   
## Call:  
## lm(formula = quality ~ log2(residual.sugar) + pH + sulphates +   
## alcohol, data = ww.test)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.96771 -0.50631 -0.03607 0.47511 2.79123   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.00406 0.86855 1.156 0.248   
## log2(residual.sugar) 0.04379 0.03131 1.399 0.163   
## pH 0.35536 0.24996 1.422 0.156   
## sulphates 0.22826 0.28725 0.795 0.427   
## alcohol 0.33738 0.03189 10.578 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7665 on 455 degrees of freedom  
## Multiple R-squared: 0.2261, Adjusted R-squared: 0.2193   
## F-statistic: 33.23 on 4 and 455 DF, p-value: < 2.2e-16

The model didn't do too bad as far as the multiple R-squared.