Predict Wine Quality Using Physicochemical Properties

By: David Yang (xdyang70@gmail.com)

Date: June 20, 2016

In this analysis report, I summarize the statistical analyses done on two public data sets (https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/). The primary goals are to answer three sets of questions. (1) What measurements are important in determining the quality of a wine? How much can we learn about quality from these measurements? (2) Are there any groups of wines that seem similar in composition and resulting quality? (3) How can we use this data to predict truly exceptional (or poor) wines? How should we use this information?

Firt, let us do some preparation jobs: set up the working directory, load in necessary libraries, and additional functions that I saved in a different R script file for convinence. You may need to save the file ("Functions Used for Wine Quality Analyses.R") onto your own computer and adjust working directory before running this Rhtml file.

```
setwd("/Users/davidy/Documents/Personal/Resume/Applications/Acorns/Output")
suppressMessages(source("Functions Used for Wine Quality Analyses.R"))
opts_chunk$set(fig.width=8, fig.height=8)
```

Now, read in the two data sets from their online sources. Check the structure of the data set to ensure they look right.

```
url1 <- 'https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-red.csv'
red_wine_original <- read.csv(url1, header=TRUE, sep=";")
str(red_wine_original)
```

```
## 'data.frame':        1599 obs. of  12 variables:
##  $ fixed.acidity       : num  7.4 7.8 7.8 11.2 7.4 7.4 7.9 7.3 7.8 7.5 ...
##  $ volatile.acidity    : num  0.7 0.88 0.76 0.28 0.7 0.66 0.6 0.65 0.58 0.5 ...
##  $ citric.acid         : num  0 0 0.04 0.56 0 0 0.06 0 0.02 0.36 ...
##  $ residual.sugar      : num  1.9 2.6 2.3 1.9 1.9 1.8 1.6 1.2 2 6.1 ...
##  $ chlorides           : num  0.076 0.098 0.092 0.075 0.076 0.075 0.069 0.065 0.073 0.071 ...
##  $ free.sulfur.dioxide : num  11 25 15 17 11 13 15 15 9 17 ...
##  $ total.sulfur.dioxide: num  34 67 54 60 34 40 59 21 18 102 ...
##  $ density             : num  0.998 0.997 0.997 0.998 0.998 ...
##  $ pH                  : num  3.51 3.2 3.26 3.16 3.51 3.51 3.3 3.39 3.36 3.35 ...
##  $ sulphates           : num  0.56 0.68 0.65 0.58 0.56 0.56 0.46 0.47 0.57 0.8 ...
##  $ alcohol             : num  9.4 9.8 9.8 9.8 9.4 9.4 9.4 10 9.5 10.5 ...
##  $ quality             : int  5 5 5 6 5 5 5 7 7 5 ...
```

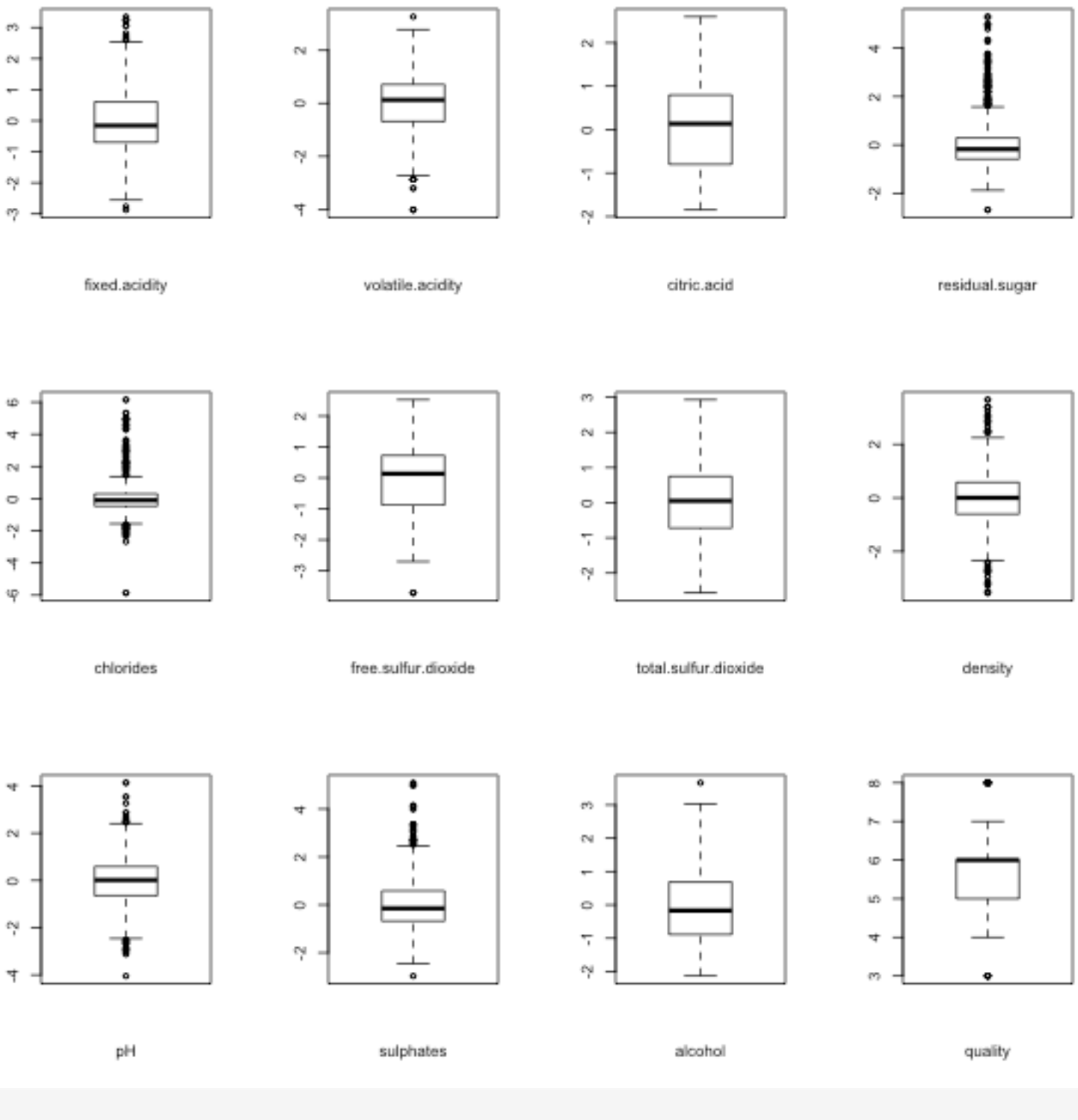
```
red_wine = red_wine_original

url2 <- 'https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv'
white_wine_original <- read.csv(url2, header=TRUE, sep=";")
str(white_wine_original)
```

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

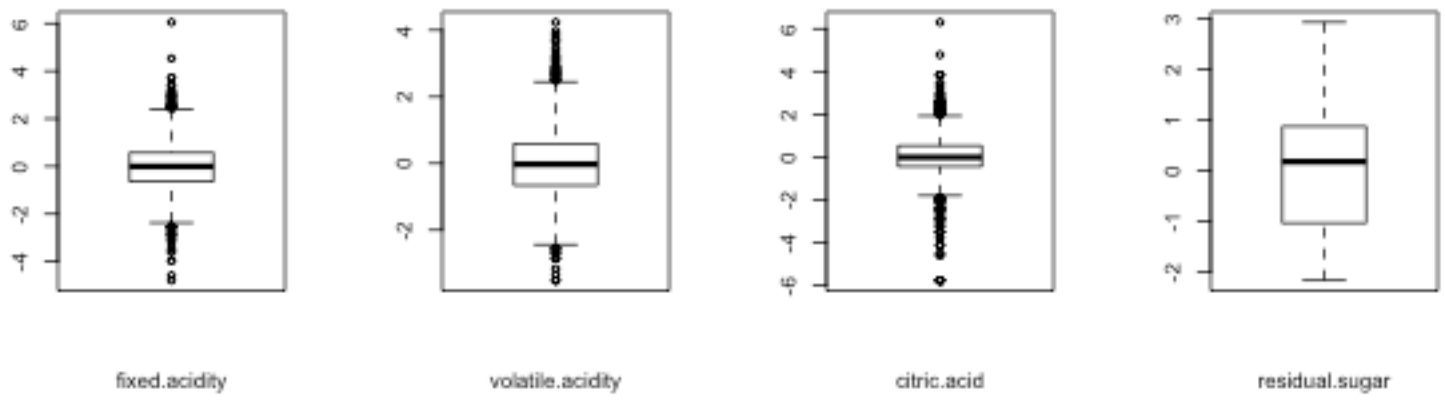
```
white_wine = white_wine_original
```

As usual, before analyzing the data, I usually conduct data exploration and apply some necessary preprocessings (e.g., transformations to remove skewness and standardization to have mean zero and stdard variance 1). Check to see if the distributions of the variables look acceptable and whether two variablces are perfectly correlated (a collinearity problem).

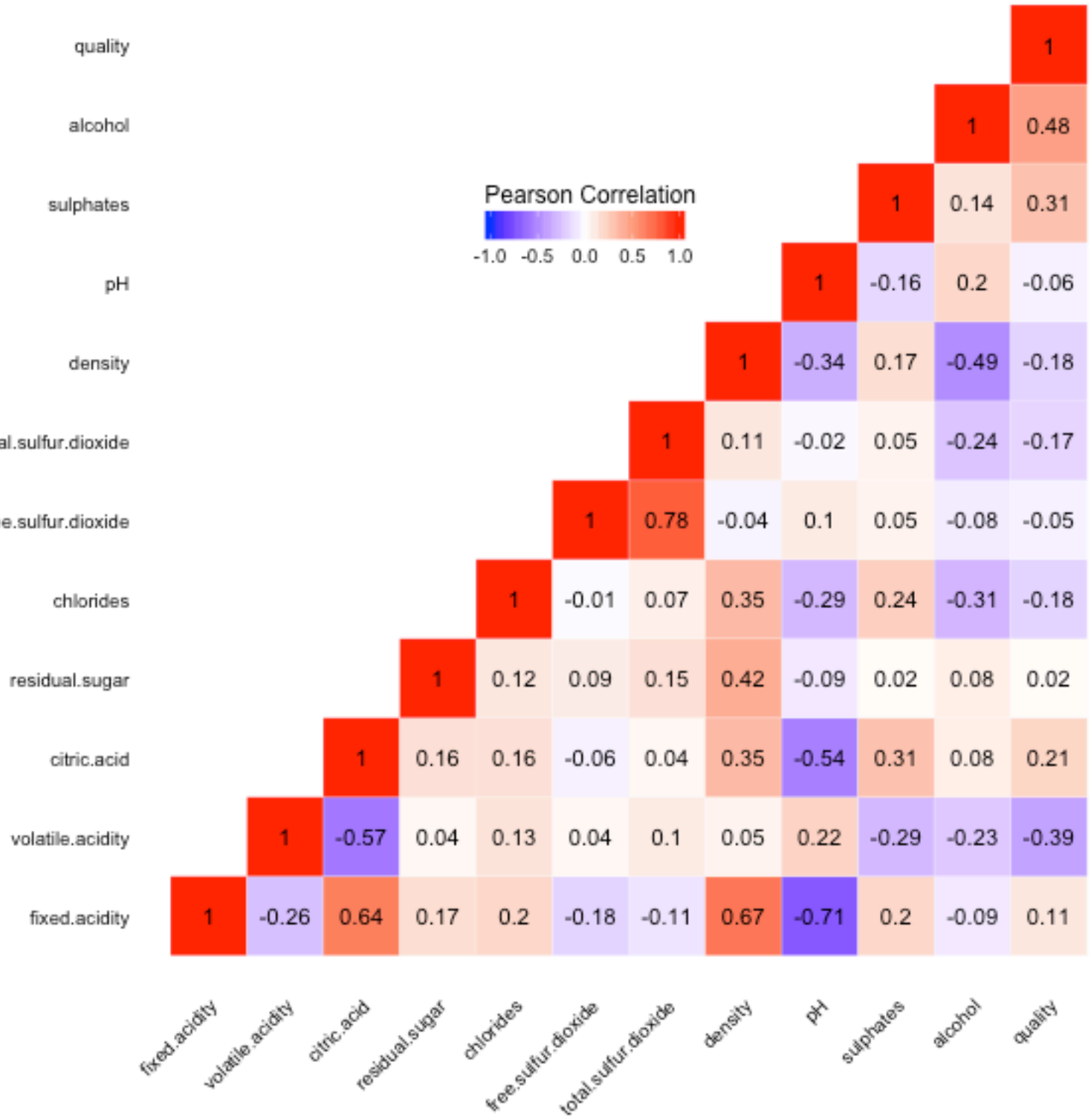
```r
# Apply transformations to make the distributions more symetric and less skewed
red_wine[,c(1:2, 4:11)] <- log(red_wine[,c(1:2, 4:11)]); red_wine[,3] <- red_wine[,3]^0.6
white_wine[, c(1:2, 4:6, 8:11)] <- log(white_wine[, c(1:2, 4:6, 8:11)]);  white_wine[,3] <- white_wine[,3]^0.45

# Standardize the variables to make them have mean=0 and variance=1
red_wine[,1:11] <- as.data.frame( scale( red_wine[,1:11] ))
white_wine[,1:11] <-as.data.frame( scale( white_wine[,1:11]))


# Check for distributions after preprocessings
draw_boxplots(red_wine, n_row=3, n_col=4, n_tot=12)    # Red Wine
```



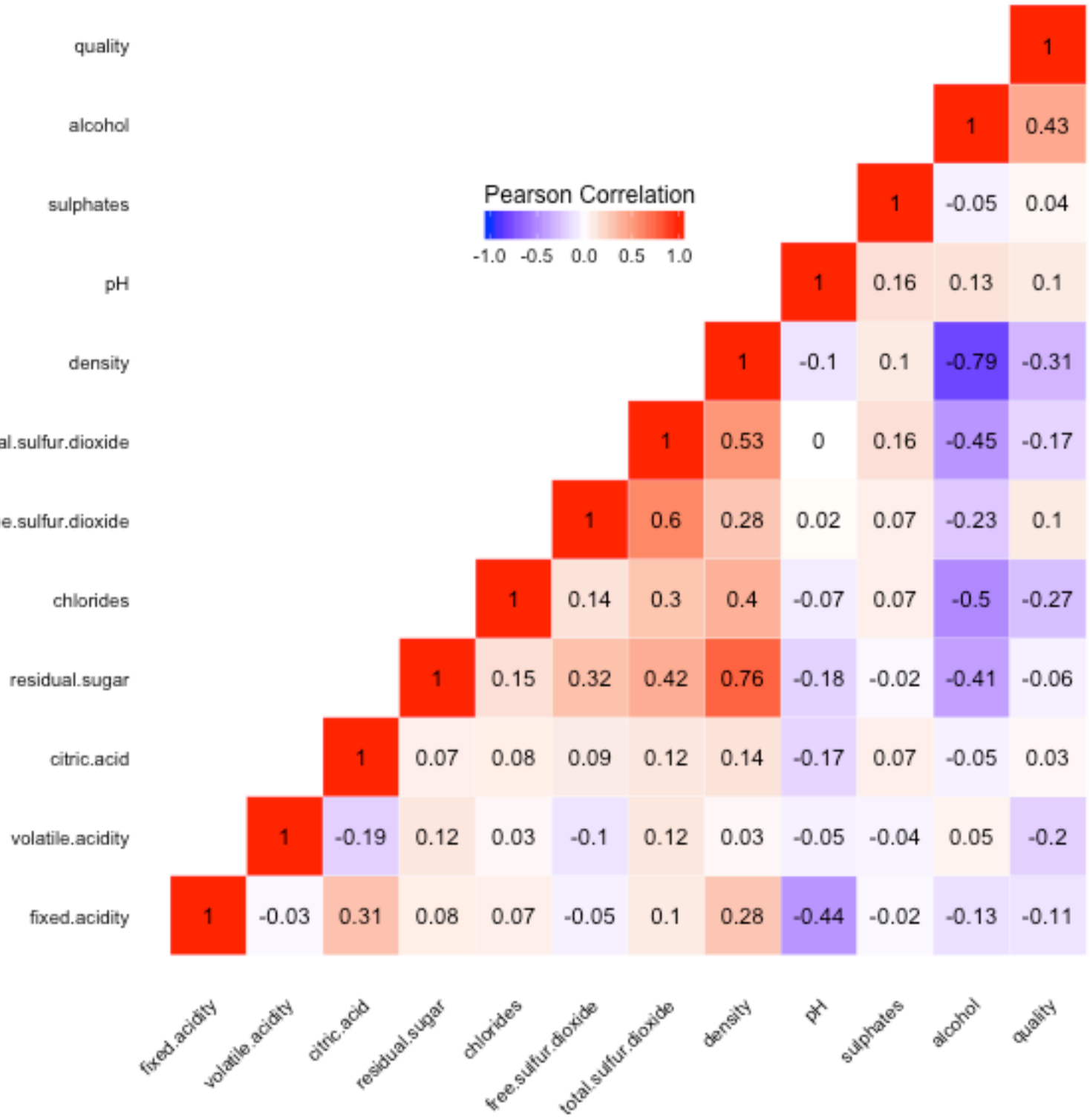

```r
draw_boxplots(white_wine, n_row=3, n_col=4, n_tot=12)  # White Wine
```




```
# Show correlation matrix after preprocessings
draw_cor_mat(red_wine)    # Red Wine
```



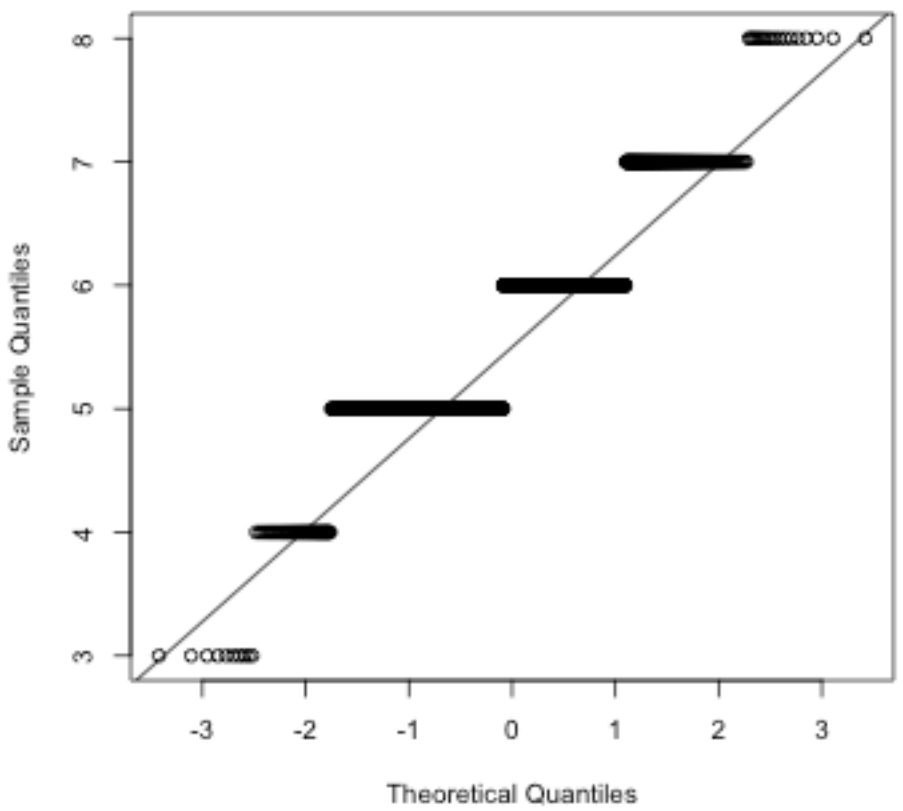

```
draw_cor_mat(white_wine)  # White Wine
```

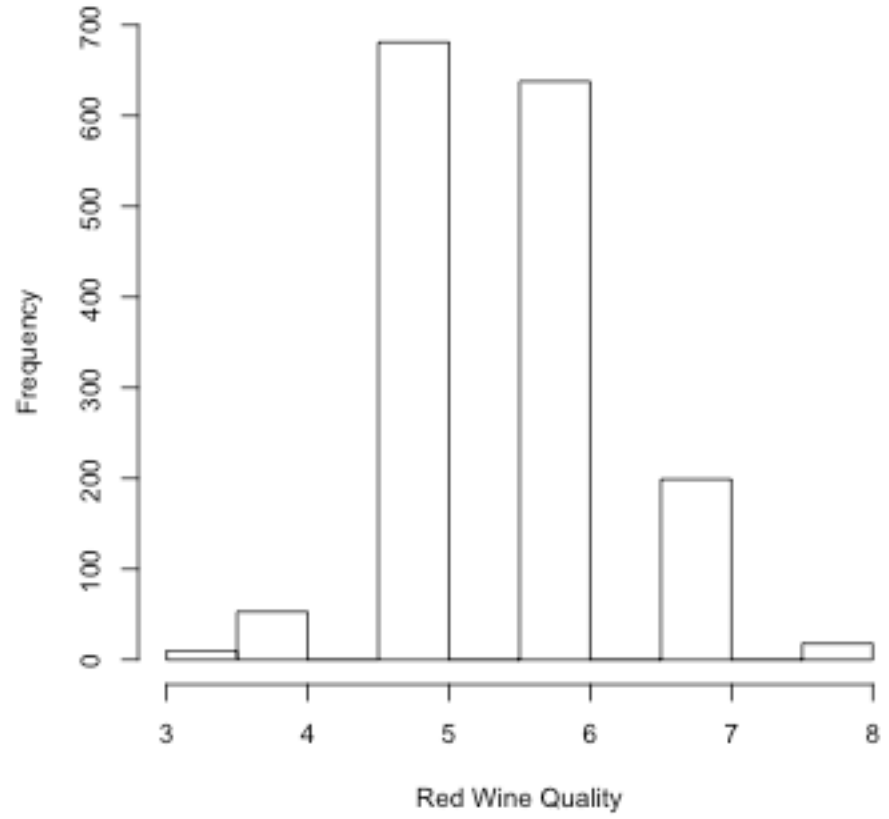


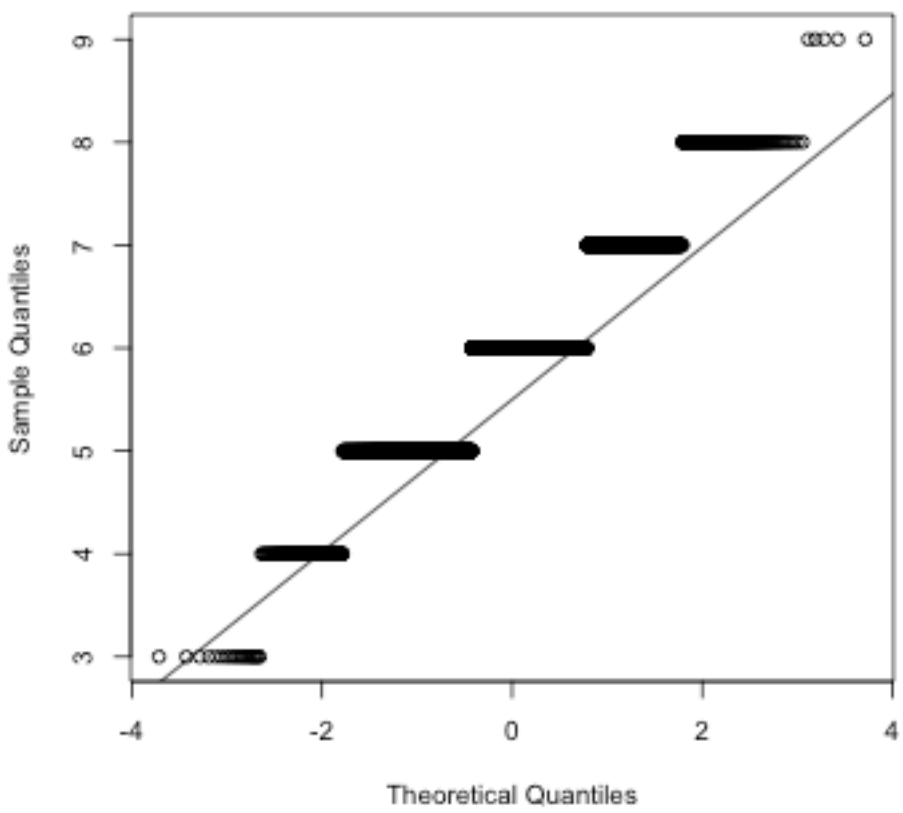
Task 1. Variable Selection & Prediction Power in Predictive Models

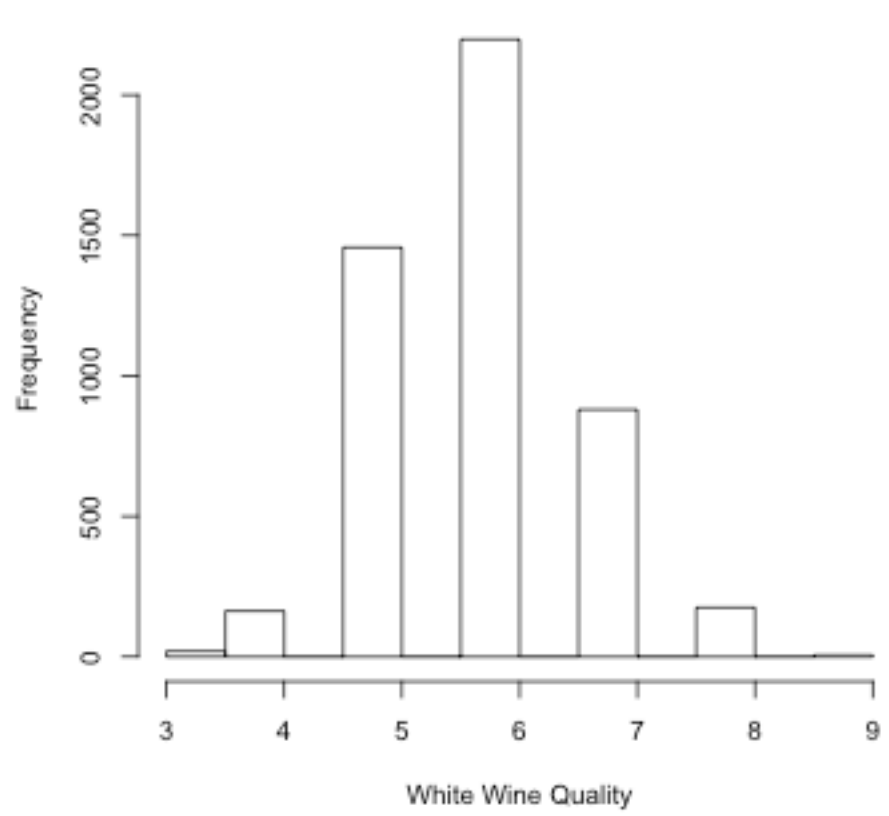
Task 1 is primarily a variable selection & prediction power assessment problem within a predictive modeling framework. There are theoretically many modeling frameworks that I can choose to figure out which subset of the 11 Physicochemical Properties (of the red and white variants of the Portuguese "Vinho Verde" wine) can be selected to predict the wine quality. Here I choose linear regression model mainly because the normality assumption looks acceptable for the distribution of *quality* (i.e., median of at least 3 evaluations made by wine experts).

```
# Check the normality assumption of Quality for Red Wine
par(mfrow=c(2,2))
# Check the normality assumption of Quality for Red Wine
check_normality(red_wine$quality, "Red Wine Quality")
check_normality(white_wine$quality, "White Wine Quality")
```









Let me first answer the first question (What measurements are important in determining the quality of a wine?). Using forward variable selection by comparing AICs, I found seven features that are statistically signifacnt (with p-value < 0.05) in predicting red wine's quality: *alcohol*, *volatile.acidity*, *sulphates*, *chlorides*, *pH*, *total.sulfur.dioxide*, and *free.sulfur.dioxide*. Thus, I rejected the null hypothesis that these input variables do not make a significantly greater than 0 contribution to the variance of the quality ranking. Adjusted R2 is 0.36, not high; but the p-value of R2 is still smaller than 0.05 and I was at least 95% confident that a linear relationship does exist between some input variables and quality ratings.

```
# Fit a linear regression model with stepwise variable selection for Red Wine
summary(fit_fwd_red <- fit_lm_step(red_wine));
```

```
##
## Call:
## lm(formula = quality ~ alcohol + volatile.acidity + sulphates +
##     chlorides + pH + total.sulfur.dioxide + free.sulfur.dioxide,
##     data = wine)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6635 -0.3577 -0.0412  0.4529  1.9140
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            5.63602    0.01618 348.365  < 2e-16 ***
## alcohol                0.29243    0.01856  15.755  < 2e-16 ***
```

```
## volatile.acidity     -0.16155    0.01832  -8.820  < 2e-16 ***
## sulphates             0.17238    0.01814   9.504  < 2e-16 ***
## chlorides            -0.08612    0.01863  -4.624 4.08e-06 ***
## pH                   -0.07630    0.01810  -4.216 2.62e-05 ***
## total.sulfur.dioxide -0.11824    0.02745  -4.307 1.75e-05 ***
## free.sulfur.dioxide   0.07968    0.02687   2.966  0.00306 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6469 on 1591 degrees of freedom
## Multiple R-squared:  0.3611, Adjusted R-squared:  0.3583
## F-statistic: 128.4 on 7 and 1591 DF,  p-value: < 2.2e-16
```

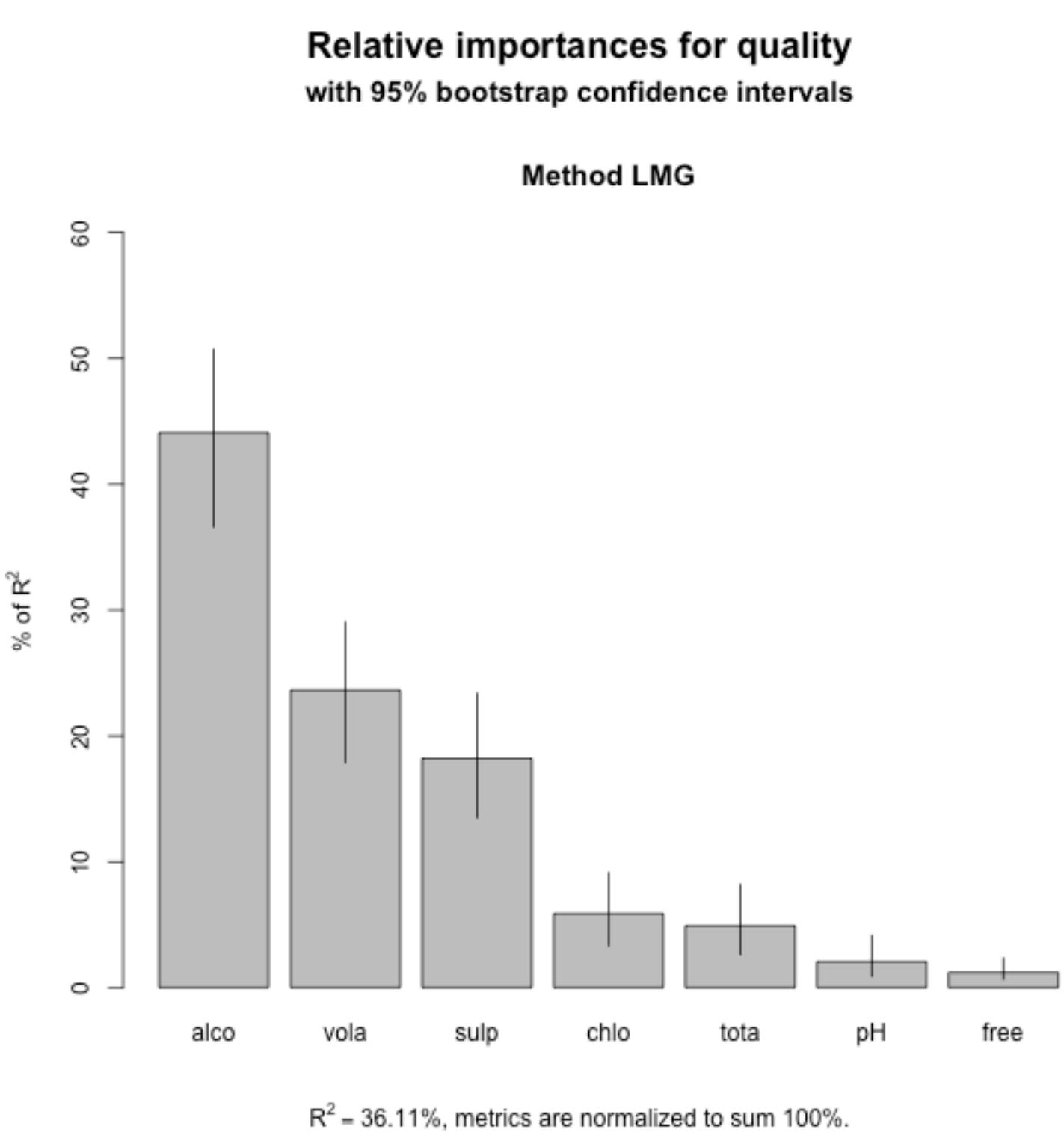
Similarly, I chose the final linear regression model for White Wine and found 10 statistically significant predictors; see the fitting model below. It is bit surprising to me that there are more singificant predictors in predcting white wine quality: *density*, *residual.sugar*, and *citric.acid* are now added to the list.

```
# Fit a linear regression model with stepwise variable selection for White Wine
summary(fit_fwd_white <- fit_lm_step(white_wine))
```

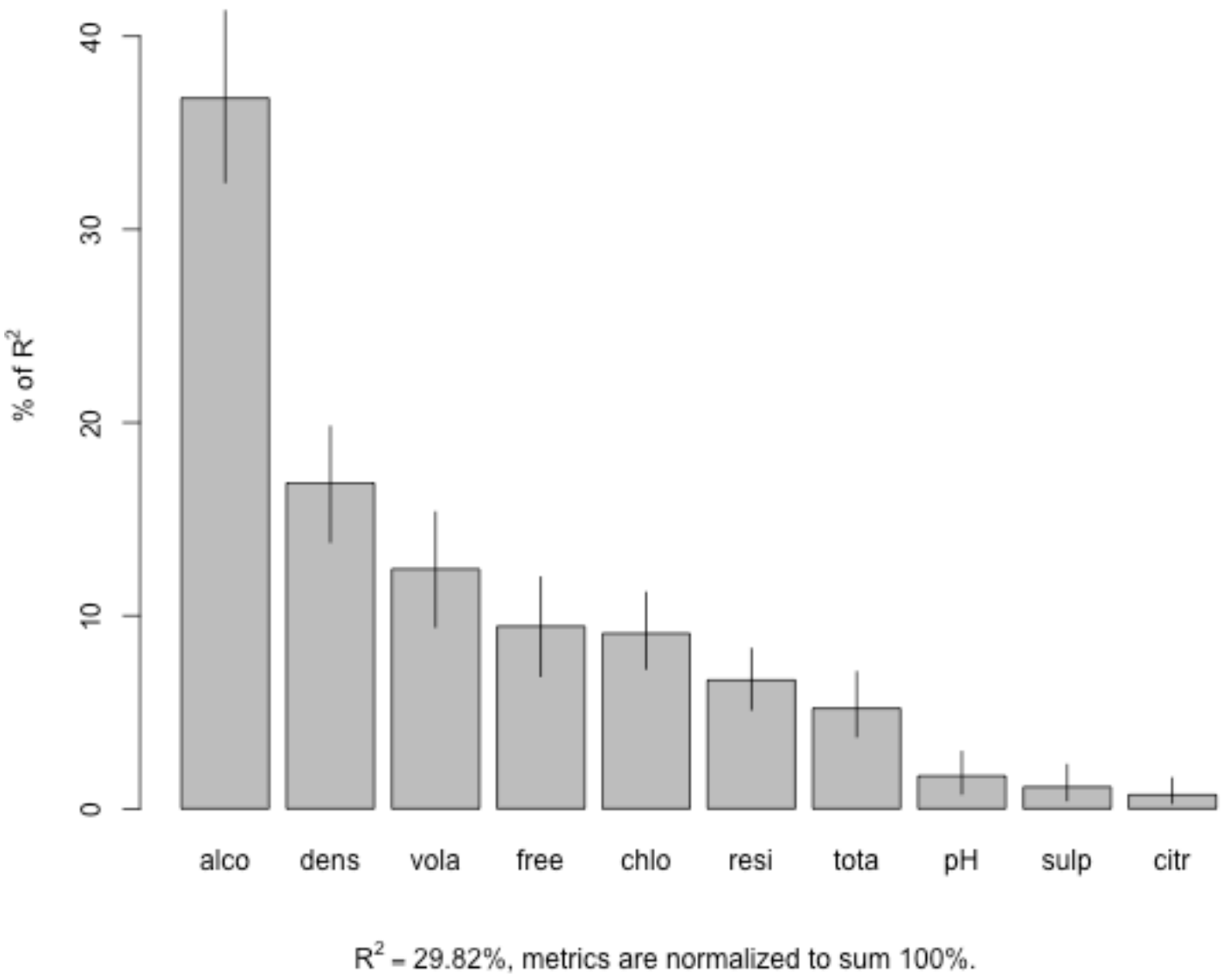
```
##
## Call:
## lm(formula = quality ~ alcohol + volatile.acidity + free.sulfur.dioxide +
##     residual.sugar + total.sulfur.dioxide + sulphates + density +
##     pH + chlorides + citric.acid, data = wine)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3789 -0.4812 -0.0373  0.4681  3.0639
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            5.87791    0.01061 553.896  < 2e-16 ***
## alcohol                0.32881    0.02127  15.462  < 2e-16 ***
## volatile.acidity      -0.17478    0.01153 -15.161  < 2e-16 ***
## free.sulfur.dioxide    0.17763    0.01394  12.738  < 2e-16 ***
## residual.sugar         0.21481    0.02013  10.669  < 2e-16 ***
## total.sulfur.dioxide -0.09630    0.01573  -6.122 9.94e-10 ***
## sulphates              0.05844    0.01110   5.265 1.47e-07 ***
## density               -0.16042    0.02940  -5.456 5.11e-08 ***
## pH                     0.04694    0.01145   4.101 4.18e-05 ***
## chlorides             -0.03727    0.01249  -2.985  0.00285 **
## citric.acid            0.02366    0.01142   2.071  0.03843 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7427 on 4887 degrees of freedom
## Multiple R-squared:  0.2982, Adjusted R-squared:  0.2968
## F-statistic: 207.7 on 10 and 4887 DF,  p-value: < 2.2e-16
```

In order to have an idea on comparing the relative predictive power of the selected seven Physicochemical Properties, I resorted to a Bootstap method in assessing the impact of each variable by excluding it from the model and see how much R2 changes. The larger drop in R2 indicates higher importance of the variable in term of its predicting power. There are other metrics used for this assessment and all suggested the consistant rankings. The following plot ranks the relative importance of the chosen physicochemical properties for red and white variants of wine. Now, it looks that *alcohol*, *volatile.acidity*, *sulphates*, *chlorides*, , and *total.sulfur.dioxide* are stonger properties than others in predicting red wine quality; while *density* and *free.sulfur.dioxide* are additional important predictors in predicting white wine quality.

```
vi_boot_red <- boot.relimp(fit_fwd_red, b = 500, type = "lmg", rank = TRUE, diff = TRUE, rela = TRUE)
plot(booteval.relimp(vi_boot_red,sort=TRUE), title="Relative Importance of Predictors for Red Wine")
```



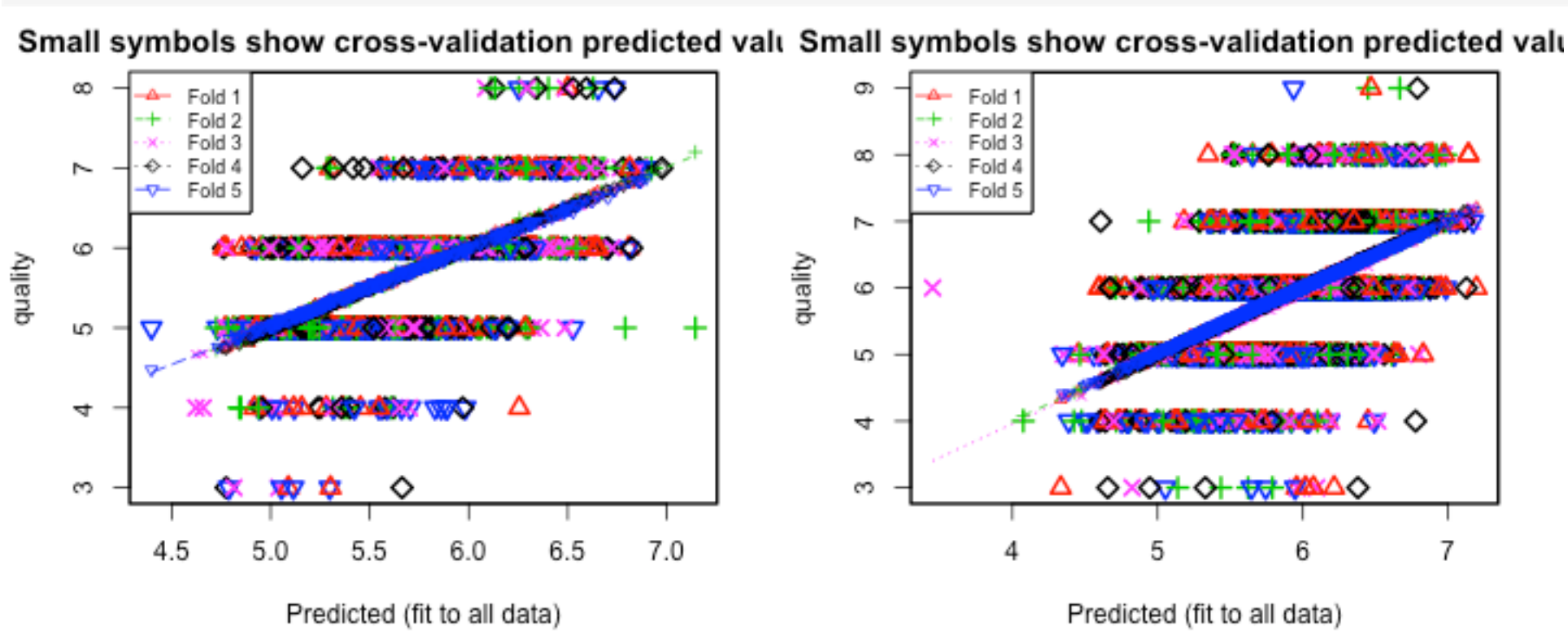

```
vi_boot_white <- boot.relimp(fit_fwd_white, b = 500, type = "lmg", rank = TRUE, diff = TRUE, rela = TRUE)
plot(booteval.relimp(vi_boot_white,sort=TRUE), title="Relative Importance of Predictors for White Wine")
```



The following analyses tries to answer the second one (How much can we learn about quality from these measurements?).

Use k-fold Cross-Validation method, we split the whole data set into k groups, then using each group as testing set to assess the prediction power of the selected model (i.e., the 7-predictor for red wine and the 11-predictor models for white wine) fitted on the rest k-1 data groups. Mean Squared Error (MSE) is a common criteria in comparing level of goodness between models.

```
par(mfrow=c(1,2))
suppressWarnings(cv.lm(data = red_wine, fit_fwd_red,  m=5))       # Overall MSE = 0.42
suppressWarnings(cv.lm(data = white_wine, fit_fwd_white,  m=5))   # Overall MSE = 0.56
```



From above cross-validation plots, we can see that the predicted tester ratings are not well aligned with the actual tester ratings, especially for ratings higher than 7 (high quality samples) and those lower than 5 (low quality samples). The MSE of the linear model for red wine is estimated as 0.42, which is fairly accuracy compared to a potential rnage scale from 1-10. The MSE for the white wine model is 0.56, which is little bit higher, but still acceptable.

```
# Prediction Error using Training-Test Split with Bootstrap resampling
  set.seed(123)
  get_accuracy_lm(fit_fwd_red, red_wine)
```

```
## [1] 0.591 0.888
```

```
  get_accuracy_lm(fit_fwd_white, white_wine)
```

```
## [1] 0.527 0.850
```

Many alternative metrics could be conceived to intuitively measure the accuracy in prediction with linear regression model. Here I introduce one set of them. If a predicted rating is within distance=d from the actual rating, then it is deemed as acceptable. Setting d=0.5 reflect the fact of rounding predicted values to closest integer numbers. Thus it is a good choice for us to see the percentage of cases in the test set whose rating is accuratly predicted using d as cutoff for rounding. When d=0.5, the accuracy is 59.1% for red wine model, and only 52.6% for white wine model. When d=1.0, then the accuracy becomes 88.8% and 85% repectively.

Task 2: Clustering & Segmentation

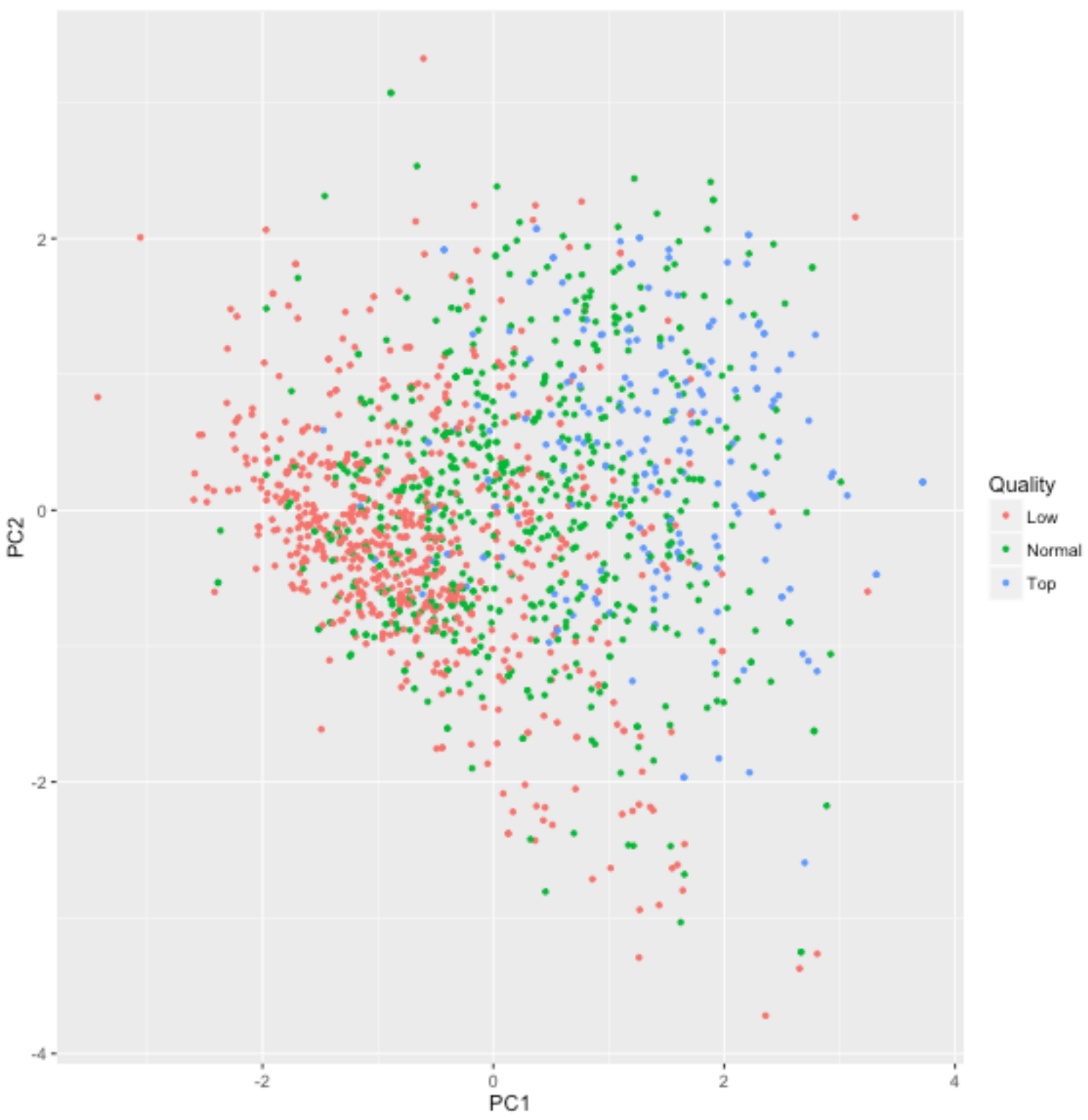
In this section, I report the custering analyses done to answer the question: **Are there any groups of wines that seem similar in composition and resulting quality?** If the question were asked as "Are there any groups of wines that seem similar in composition", then I would have chosen any unsupervised clustering methods to do the job using all 11 physicochemical properties. Nonetheless, the question contains "...and resulting quality", which implies that we may not need to bindly put all 11 variables into clustering because not every property is useful in predicting the wine qulity. Thus, in the following, I only show the clustering analyses based on relatively important predictors.

To make my job easier, I first categorize samples into three groups based on quality ratings: Low (5 or lower), Normal (6), and Top (7 or higher), each is assigned a distinct color for visual inspection. For Red Wine data, I chose the top 3 predictors and for White Wine, top 5.

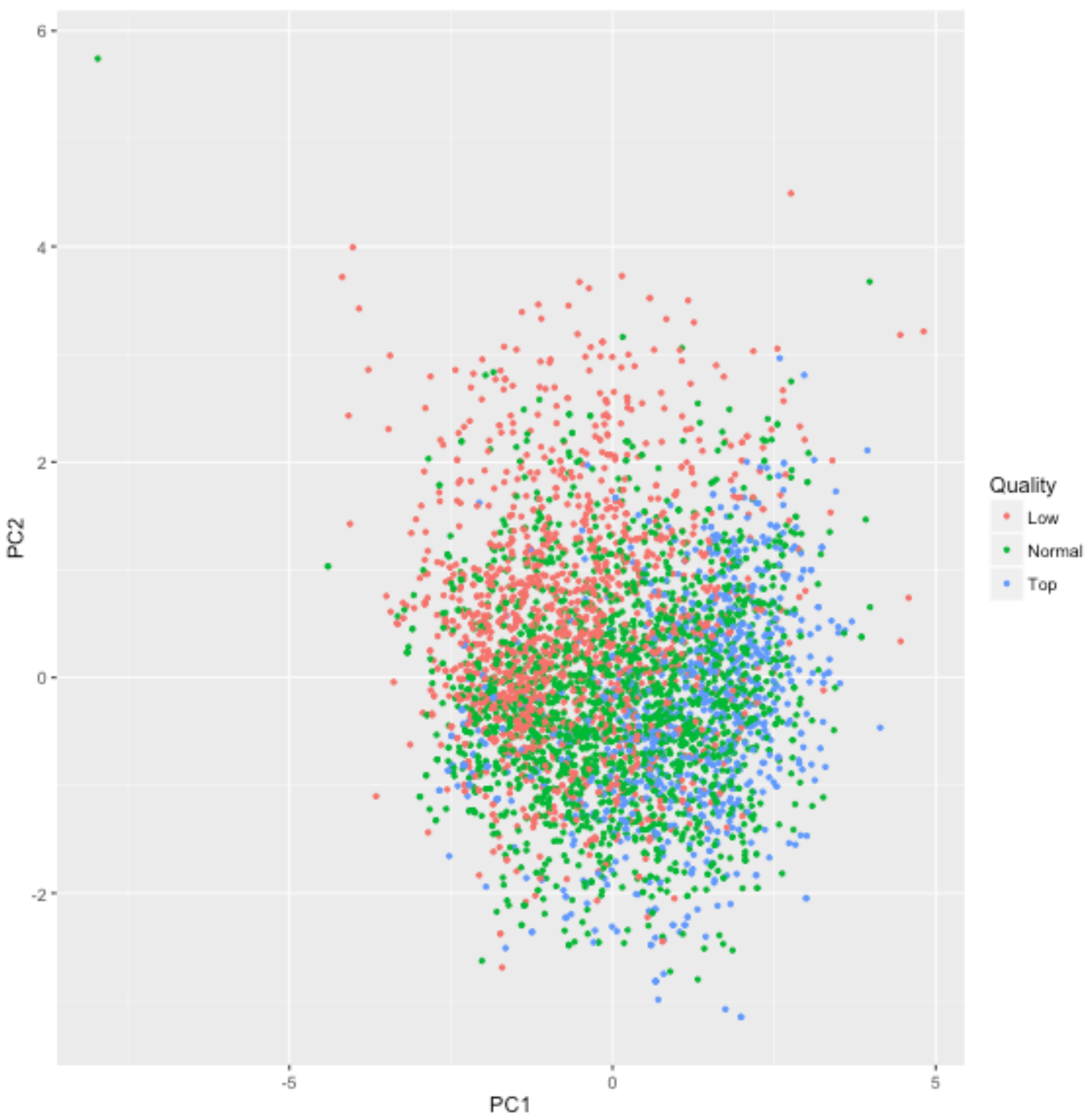
```
## Categorizing Quality into 3 Levels: -1 (bad), 0 (normal), +1 (good) and set 3 colors
red_wine_sub <- get_subset(red_wine, "Red")$sub
red_quality_col <- get_subset(red_wine, "Red")$col
white_wine_sub <- get_subset(white_wine, "White")$sub
white_quality_col <- get_subset(white_wine, "White")$col
```

When high dimensional data are given, one typical strategy resport to dimensional reduction techniques. Principal component analysis (PCA) is a popular choice for this. By choosing the top 2 prinicipal components (PC1 and PC2), and show the scatter plot of them with each point colored by one of three options defined for Low, Normal, and Top quality levels. I aimed to see whether the three quality groups can be clearly detected in the 2-dimensional space.

```
  ## PCA (principal Component Analysis)
  prin_comp_anal(red_wine_sub, red_quality_col, 0.9)        # Red Wine
```



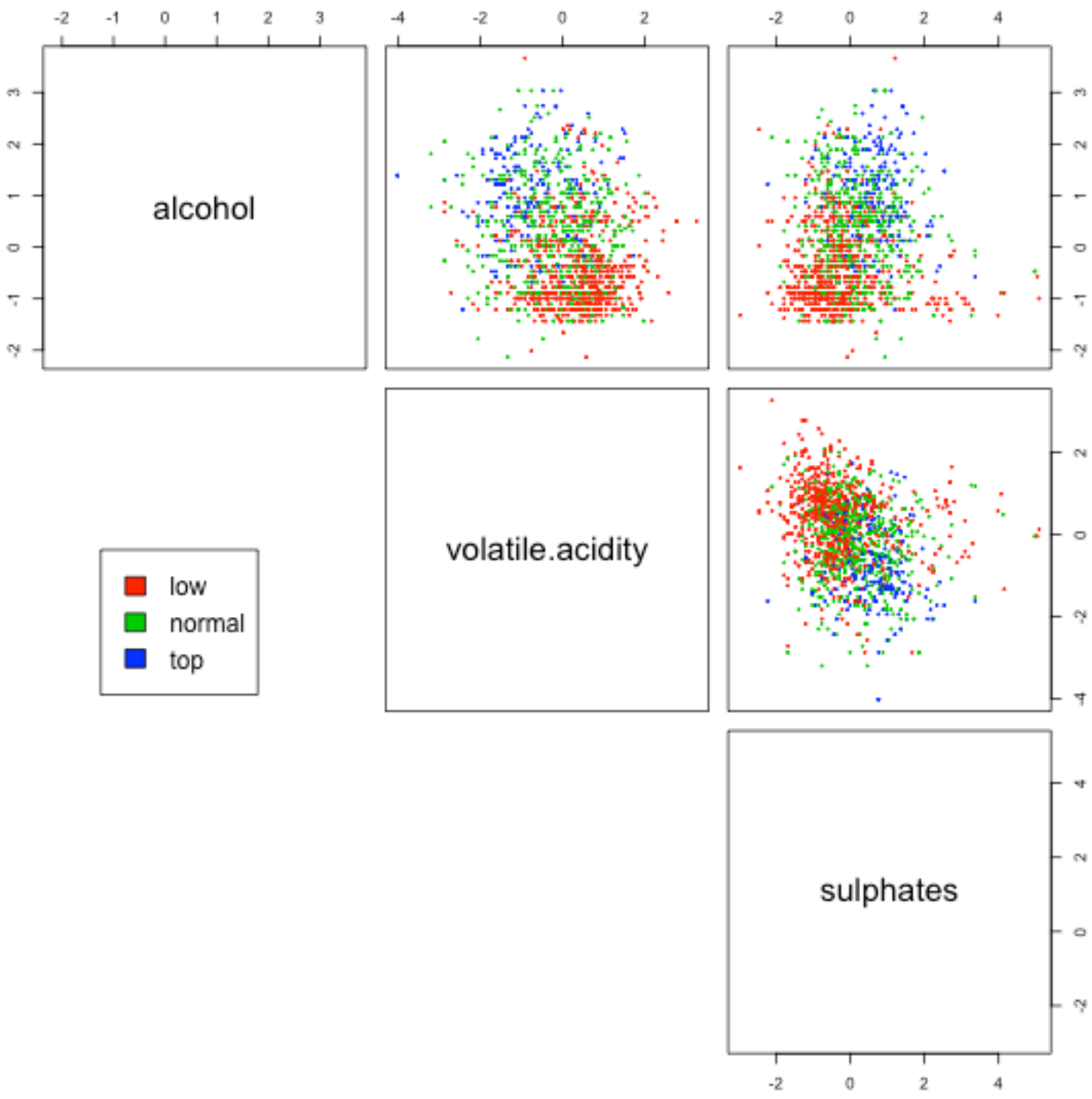

```
prin_comp_anal(white_wine_sub, white_quality_col, 0.8)     # White Wine
```



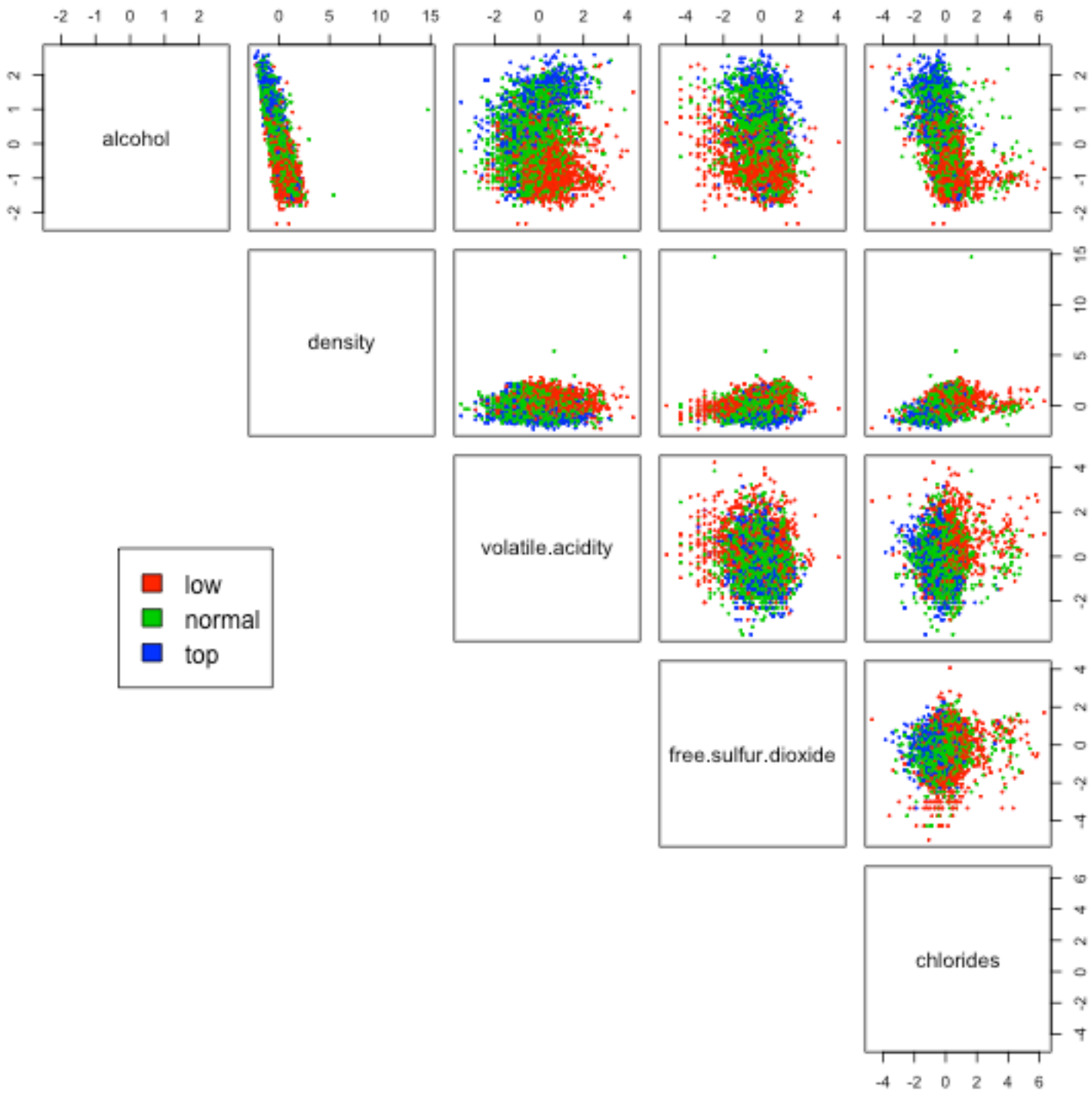
Compare the above scatter PCA plots, we can see that the three classes are fairly overlapped, especially for the white wine samples. This is consistent to the above regression analysis results. But, we do see some good evidence in suggesting calssifying wine samples into similar groups based on the choosen set of relatively important variables.

A limitation of PCA is its less intuitive interpretation becasue all the factors are weightedly summed into subspace. To observe membership patterns in the original n-dimensional space (n=3 and 5 here), once again, many plotting strategies may exist, athough I only depict two of them below. The first option is pairwise scatter plot matrix, which shows how quality groups are distributed in the 2-dim space defined by any pair of features. By breaking the n-dim space into 2-dim marginal spaces, we lose joint distributionnal patterns.

```
## Pairwise scatter plot
pairwise_scatter(red_wine_sub, red_quality_col)
```

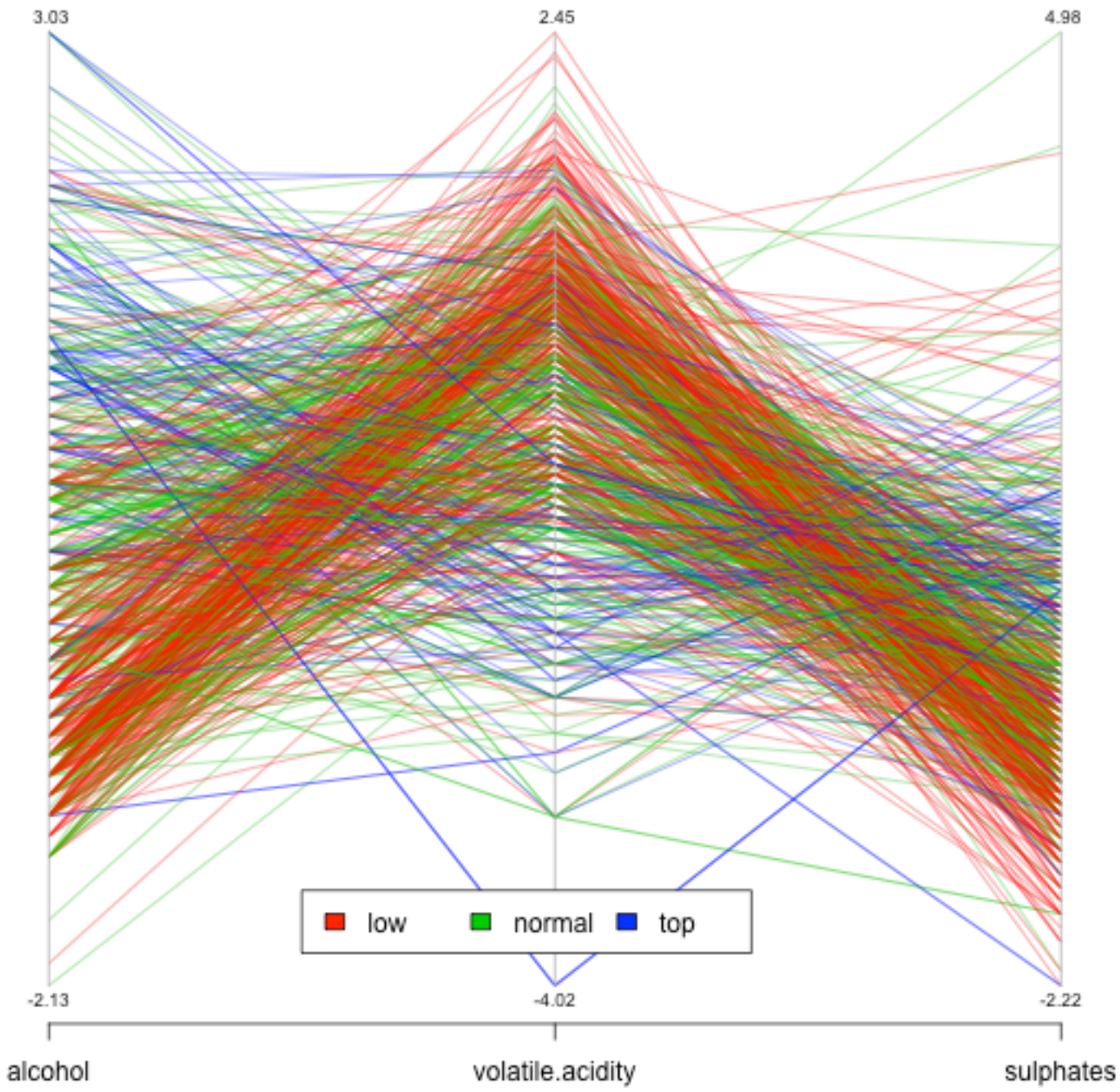



```
pairwise_scatter(white_wine_sub, white_quality_col)
```

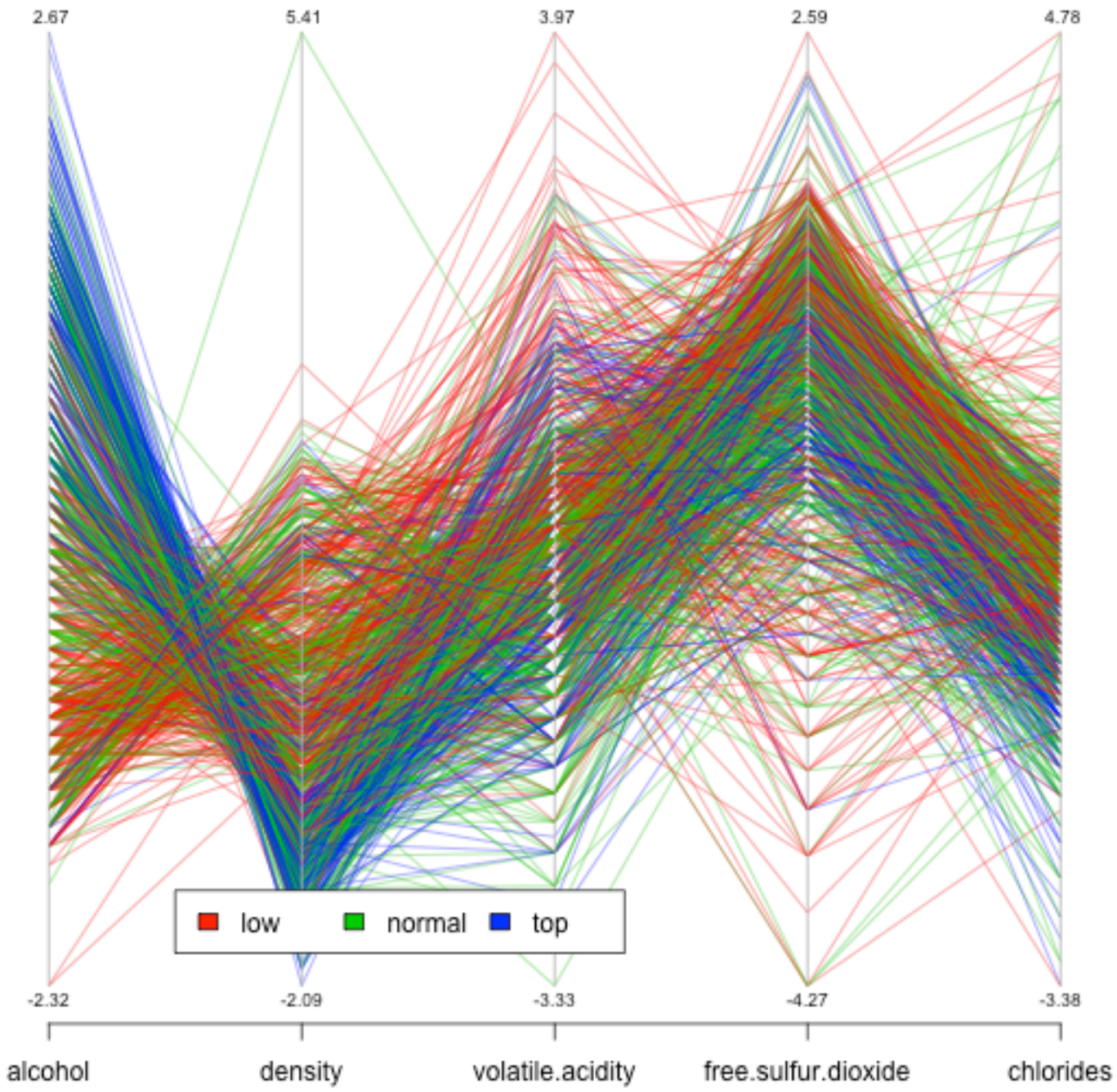


To solve the above limitation, I choose parallel coordinates plot to visualize the joint distributions of n features (n=3 for red and 5 for white samples) through the linked lines, each is colored to show categories. From this plot, we can clearly see how difference samples (colored lines) could be clustered based on values of physicochemical traits. For example, Red Wine samples tend to have low quality is they have relatively low level alcohol and sulphates, but high volatile acidity.

```
## Parellel Coordinates Plot
parallel_coordinate_plot(red_wine_sub, 0.5, red_quality_col)
```




```
parallel_coordinate_plot(white_wine_sub, 0.2, white_quality_col)
```



Task 3: Classification and Prediction

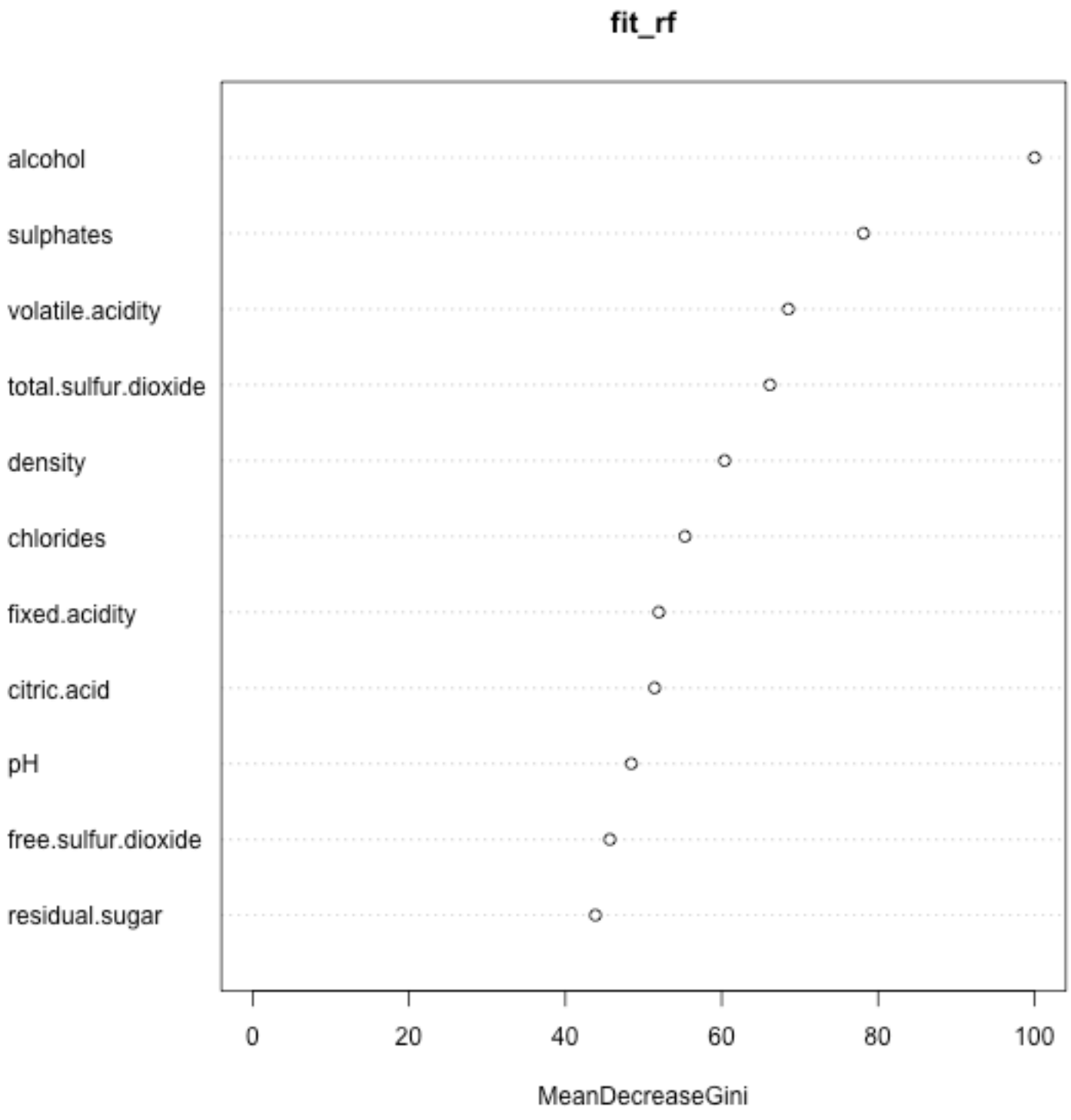
In this last section, I will answer the last two questions mainly using Random Forest (RF), an ensemble methods for making predictions by employing many small models (i.e., decision trees with small number of nodes). We choose RF because my personal expereince suggests this method has the best classification accuracy.

(A) How can we use this data to predict truly exceptional (or poor) wines?

Again, based on the frequencies of the samples across quality rating levels, we set cutoffs to end up with top (rating>6), normal (rating = 6), and low (rating <6) qualities. Of cause, I can change the cuttoffs to redefine exceptinal good or bad quality wines. When fit the FR, we would include all 11 predictor variables into the scope of building decisional trees. This is for two reasons: (1) RF can handle big number of variabls by only choosing randomly a small subset of predictors into building tees; (2) we want to use RF to rank again the importance of physicochemical properties to check if they are consistent to the performance of linear regression.

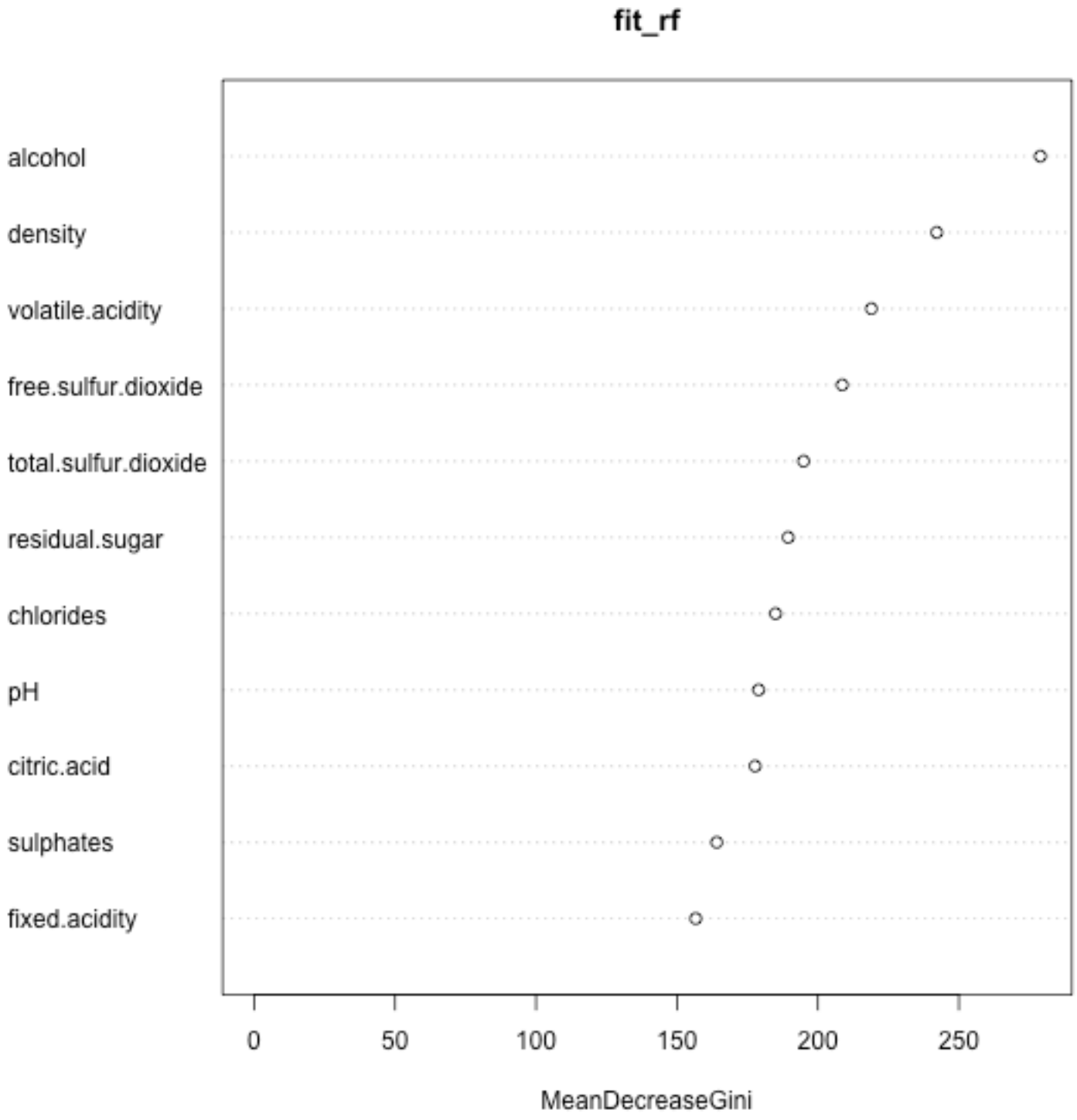
```
##  Random Forest for predicting truely exceptional (or poor) wines
set.seed(1)
fit_rf(red_wine, 1000)     # Red Wine
```

```
##
## pred      low normal top
##   low     179     54   2
##   normal   26    136  23
##   top       1     10  49
## [1] 0.758
```




```
fit_rf(white_wine, 1000)   # White Wine
```

```
##
## pred      low normal top
##   low     334     94  10
##   normal 140    515 104
##   top       6     64 203
## [1] 0.716
```



From the above fitted RF model with all predictors, we see that the estimated classification accuracy is 75.8% and 71.6%, respectively, for Red and White wines. This classification rate looks fairly good. Varible importance is also depicted above. Note that in this secition RF works with categized quality (low, normal, and top), while the original rating scales were direcly applied in Task 1, thus the two models (RF vs Linear Regression) are not absolutely comparble. In RF we see some evidence that *desity* becomes a more predictive of quality level for Red Wine samples.

(B) How should we use this information?

Once we have some idea on which of the 11 physicochemical properties are more strongly correlated with the winw quality ratings, then we can use them to serve us on many directions: e.g., (For example, (1) we can use the list of strong predictors jointly into a model (e.g., RF, linear regression, or logistic regression models) to predict the quality of a specific variant of red/white wine; (2) one can better control the quality of wines during their whole brewing processes.