PROJECT ON

“CASE STUDY OF WINES”

BY

ASHISH KUMAR MISHRA

Under the esteemed guidance of Shri. Muquayyar Ahmed,

Mini Project: Case Study on Wines

Objective

To build a classification model which will classify the quality of wine depending on multiple factors.

Parameters

- Alcohol
- Chlorides
- Citric acid
- Density
- Fixed acidity
- Free sulfur dioxide
- pH
- **Quality** (target)
- Residual sugar
- Sulphates
- Total sulfur dioxide
- Volatile acidity

Alcohol

It is the alcohol content in the wine. It is a good predictor of wine quality. Quality of wine tends to increase with alcohol content up to a certain limit.

Chlorides

Chlorides can come from grapes used for fermentation. It also depends on the soil. Soils with more irrigation tend to be more saline. This adds to the savory flavor of wine. At the same time, too much chloride intake adversely affects health. For the same reason, many wine-producing countries have (widely varying) legal maximums for sodium chloride.

Citric Acid

Citric acid plays a major role in the acidity of wine. It can give the wine more "freshness" and will effectively make the wine more acidic. However, bacteria use citric acid in their metabolism, thus the citric acid added may just be consumed by bacteria, promoting the growth of unwanted microbes. So, it has both advantages and disadvantages, hence, it is essential to get the right amount of citric acid.

Density

It depends on the water and sugar content of the wine. It has a high correlation with alcohol.

Fixed Acidity

Acids contribute majorly to the taste. The sourness or tartness that is a fundamental taste of wine is provided by acidity. Wines lacking in acidity are "flat". But increase the fixed acidity too much and the wine would be too sour, which might not be to everyone's liking.

Free Sulfur Dioxide and Total Sulfur Dioxide

Total SO_2 = Free SO_2 + Bound SO_2

Bound SO_2: are the sulfites attached to either sugars, acetaldehyde or phenolic compounds Sulfur dioxide is one of the most effective tools that a winemaker has to protect wine and influence what it will taste like. SO2 is a by-product of fermentation too! Since, some volume of SO_2 is already bound, hence it cannot be used to protect wine from oxidation, unlike the free SO_2. Meaning the bound SO_2 is not very useful in determining the quality of wine. Hence, free SO_2 is a more useful factor in determining wine quality. As some people are allergic to bisulfites, there are some organic wines as well, which do not contain any added sulfur (but they do contain bound bisulfites). SO_2 in its gaseous form, or molecular SO_2, is very effective at killing microbes in wine. Free SO_2 also helps in keeping the wine fresh by binding with aldehyde in the oxidation product, forming a harmless and odorless molecule.

pH

The pH scale technically is a logarithmic scale that measures the concentration of free hydrogen ions floating around in wine. The stronger the acid, the more hydrogen ions it will have. So, in essence, it is a measurement of how strong an acid is.

Quality

The quality of wine as judged by the connoisseur.

Residual Sugar

Residual Sugar refers to any natural grape sugars that are left over after fermentation ceases (whether on purpose or not). The juice of wine grapes starts out intensely sweet, and fermentation uses up that sugar as the yeasts feast upon it. The sweet taste is essential for balancing the acidity in wine. Wrong proportion may hamper the taste of wine. Hence, it needs to be just right.

Sulphates

It is a wine additive which contributes to the SO_2 levels. It is added by the manufacturer.

Volatile Acidity

It's basically the process of wine turning into vinegar. It imparts a vinegary taste and aroma to the wine. The net effect is undesirable. SO_2 is critical to controlling the bugs that cause volatile acidity. Quality of wine decreases with increase in volatile acidity.

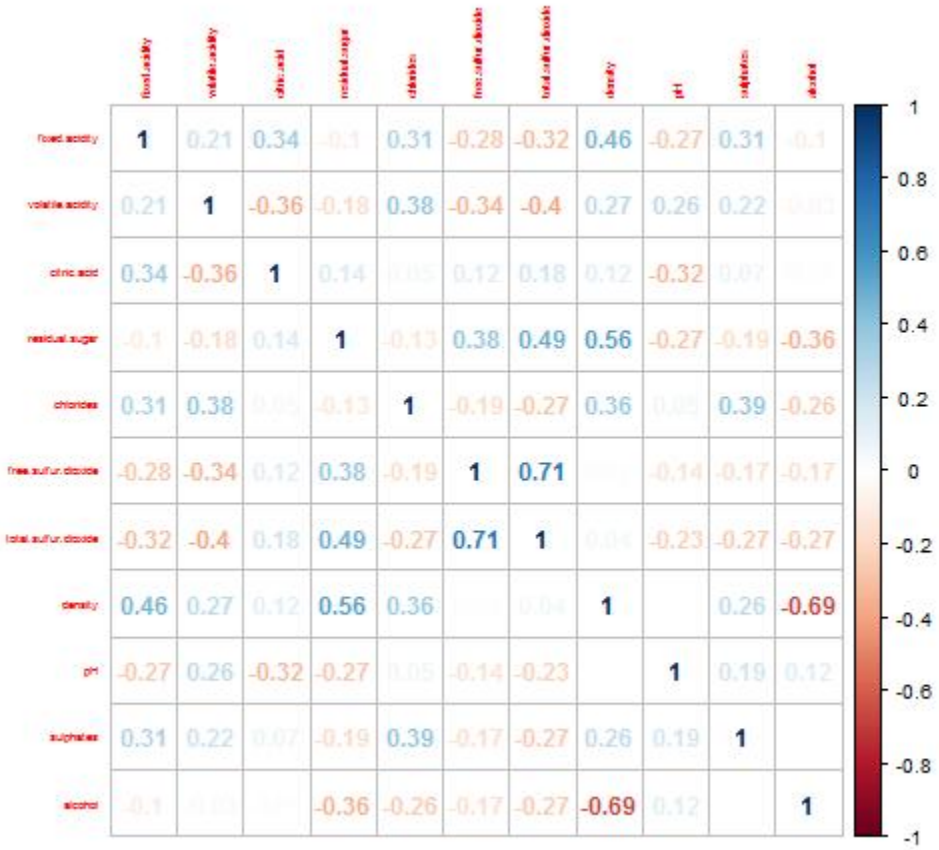
Quality

It is Output variable (based on sensory data).

Data Pre-Processing

Data is read from csv files and data of both, red and white, wines is merged into a single data frame. Data type is changed to numeric for fields imported as char. Then a field of “color” is added to distinguish between red and white wines. Color is treated as a level. All fields except “color” and “quality” are normalized and extreme outliers removed.

Feature Selection



Strong correlation between:

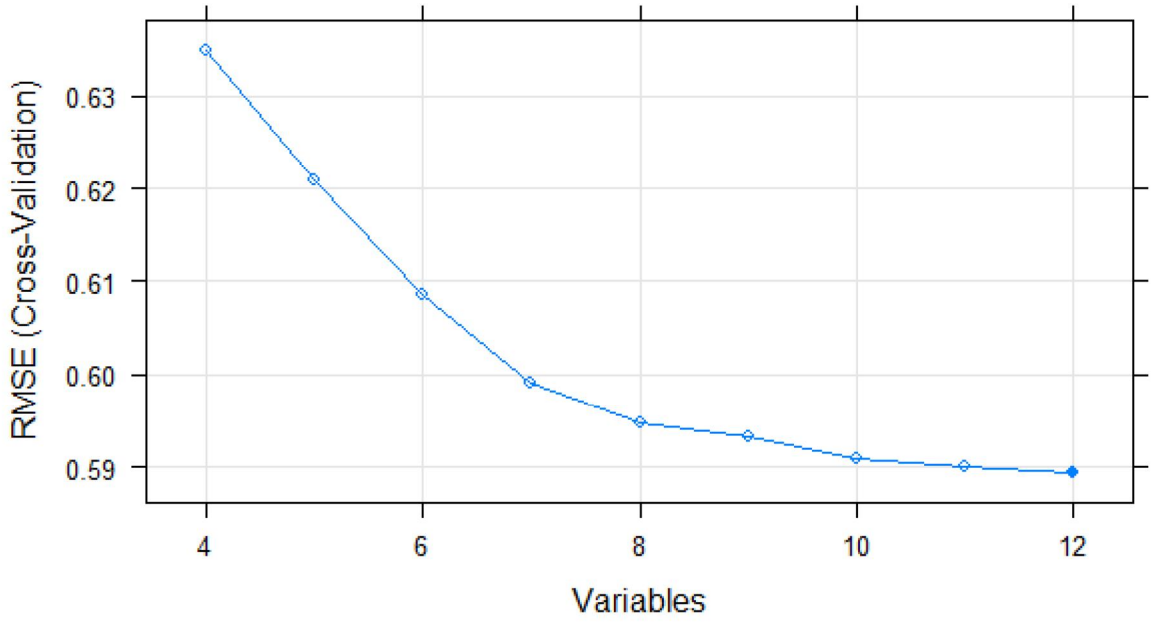
- *total sulfur dioxide* and *free sulfur dioxide* **,** Positively related. Both increase with each other.

Suggestive correlation between:

- *density* and *residual sugar* **.** Both increase with each other.

- *density* and *alcohol* . Negatively related. Decrease with increase in another.

RFE (Recursive Feature Elimination) Results



The above plot shows that the optimum number of variables for our predictions should lie between 6 to 8. Adding more features will add to the complexity of our model by marginally improving accuracy. And in such cases, model with less complexity is preferable over the complex model with marginal improvement in accuracy. The error metric used here is RMSE (root mean square error). The method used here is cross validation.

Using our intuition based on the given features and Recursive Feature Selection by elimination (RFE), we can deduce the following:

- Color is the least important parameter in determining wine quality.
- Alcohol and residual sugar correlate with density and density is the least effective parameter.
- Fixed acidity and pH tell the same thing and out of those two, pH is a better indicator.
- Total sulfur dioxide includes both bound and free sulfur dioxide. As explained above, bound is of no use to us and we already have the measure of free sulfur dioxide.

RFE results RFE gives us the five most important features needed, ranked on the basis of their impact. The features are thus ranked on their impact as follows:

1. Alcohol
2. Volatile Acidity
3. Free Sulfur Dioxide
4. Sulphates
5. pH
6. Residual Sugar

The first five are strong predictors of wine quality, while pH gives a noticeable edge in our predictions. Using the RFE results that we had, we find that the optimum number of variables lies between 6 to 8. The difference in the accuracy being very small. Apart from the above mentioned 6 features, total sulfur dioxide and fixed acidity improve the predictions marginally but add to the complexity of the model, hence they should not be used in building the model. Density, chlorides and citric acid had the least impact, so they should be discarded as well for modelling.

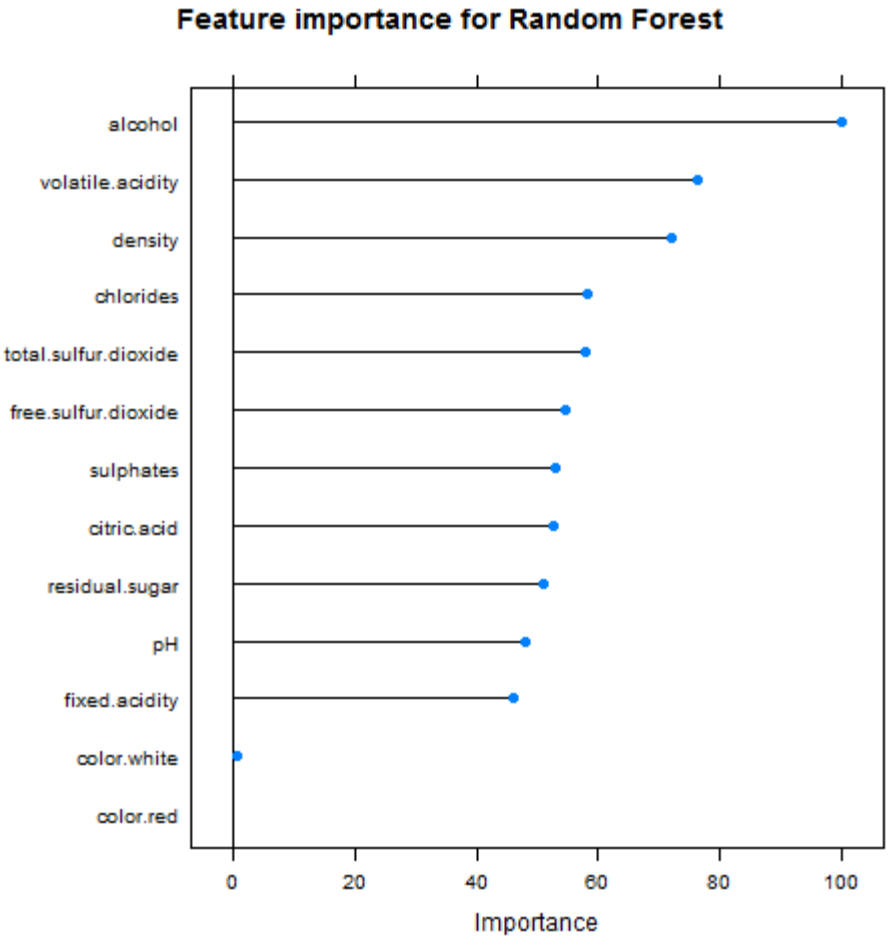
Modeling

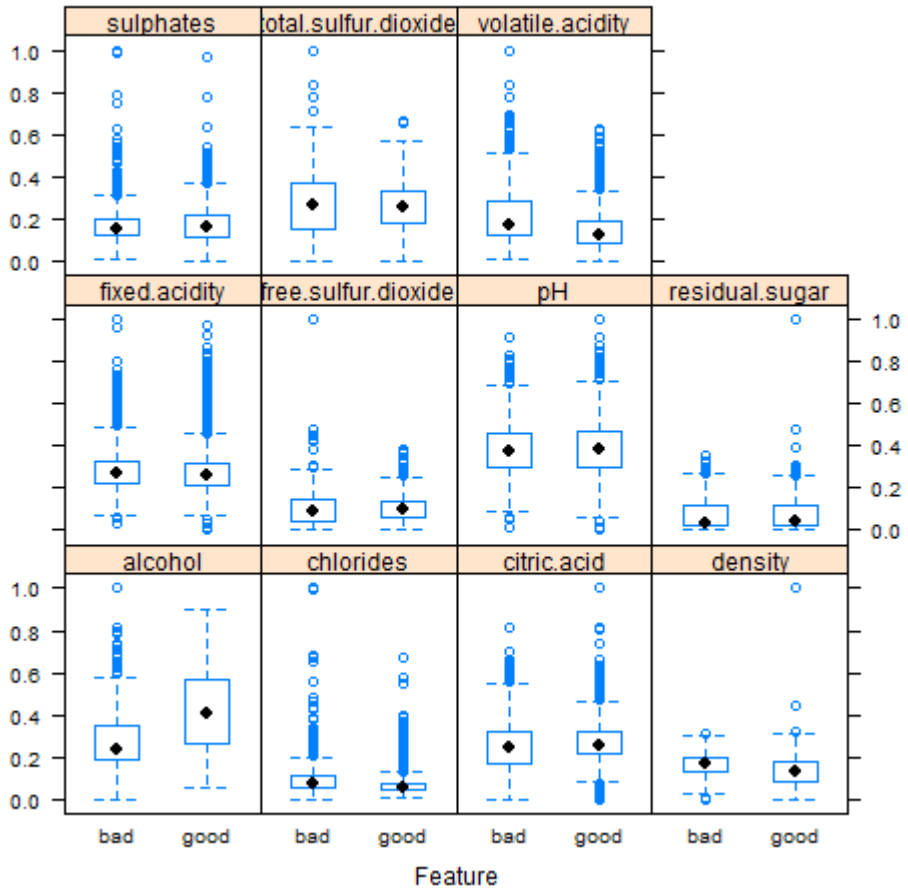
A random forest model is built using all the features and all the features are ranked

The plot of Error vs Trees suggests that increasing the number of trees beyond 100 doesn't help much.

Adding a large number of trees unnecessarily would only add to the complexity of the model, resulting in slower models.

Here’s the zoomed in image:





So the final random forest model has 1000 trees.We have used all features in train as well as test data set. The data was split into 60% training data and 40% test data. Then

using these parameters, a random forest was simulated and the accuracy was determined as 0 units of difference from the actual value. The process was done using all the features and the accuracy was around 82.75%.

Inference

- Alcohol is a strong predictor of wine. The average alcohol content of good wines (quality greater than 6) is 11.4%. Minimum alcohol content is 8% for good wines.
- Wine quality decreases with increase in volatile acidity. Good quality wines have around 0.3 units of volatile acidity on average.
- Wines need freshness and quality tends to increase with increase in amounts of citric acid and pH up to a certain level. Then it decreases. Citric acid levels should not exceed 0.5 units for a good wine and pH should be under 3.8. Fixed acidity levels should be around 6 to 10 units.
- Residual sugar levels should be around 5-10 units and no more than 20 for a good wine. Meaning the wine should not be too sweet.
- Chloride levels should be under 0.05 units and wine quality decreases with much increase in chloride levels, meaning salty wines are not preferable.
- Free sulfur dioxide levels tend to be variable anywhere from 3 to 100 for a good wine but tends to decrease wine quality if over 100. Total sulfur dioxide levels tend to be higher than 100 units but less than 200 for really good wines with quality greater than 7. Sulphates tend to be between 0.4 to 1.0 units. This means that sulphates should be added in moderation.
- Density tends to be around 1 units and doesn't seem to affect wine quality much.

More Inferences

Analyzing the data we can come up the following conclusion:

1. When alcohol percentage decreases, density grows.
2. In general alcohol level of red wine is higher than alcohol level of white wine.
3. When fixed acidity increases density of red wine increases as well. White wine almost doesn't show any correlation.
4. Total sulfur dioxide and level of residual sugar are positively correlated. Correlation shows higher value with white wine.
5. White wine density and residual sugar level have positive correlation.
6. Alcohol level of white wine decreases with the growth of residual sugar level.
7. Mostly frequent quality levels of red and white wine are 5 and 6.

Code:

```
rm(list=ls())
library(corrplot)
library(caret)
```

```
library(randomForest)
library(gridExtra)

library(doSNOW)
registerDoSNOW(makeCluster(3, type = 'SOCK'))
today <- as.character(Sys.Date())
setwd("K:/edwisor/CaseStudyOnWines/CaseStudyOnWines/")
white <- read.csv('winequality-white.csv', sep=';')
red <- read.csv('winequality-red.csv', sep=';')
red[, 'color'] <- 'red'
white[, 'color'] <- 'white'
df <- rbind(red, white)
attach(df)
#data plot
par(mfrow=c(2,2), oma = c(1,1,0,0) + 0.1, mar = c(3,3,1,1) + 0.1)
barplot((table(quality)), col=c("slateblue4", "slategray", "slategray1", "slategray2", "slategray3", "skyblue4"))
mtext("Quality", side=1, outer=F, line=2, cex=0.8)
library(MASS)
truehist(fixed.acidity, h = 0.5, col="slategray3")
mtext("Fixed Acidity", side=1, outer=F, line=2, cex=0.8)
truehist(volatile.acidity, h = 0.05, col="slategray3")
mtext("Volatile Acidity", side=1, outer=F, line=2, cex=0.8)
truehist(citric.acid, h = 0.1, col="slategray3")
mtext("Citric Acid", side=1, outer=F, line=2, cex=0.8)
par(mfrow=c(1,5), oma = c(1,1,0,0) + 0.1,  mar = c(3,3,1,1) + 0.1)
boxplot(fixed.acidity, col="slategray2", pch=19)
mtext("Fixed Acidity", cex=0.8, side=1, line=2)
boxplot(volatile.acidity, col="slategray2", pch=19)
mtext("Volatile Acidity", cex=0.8, side=1, line=2)
boxplot(citric.acid, col="slategray2", pch=19)
mtext("Citric Acid", cex=0.8, side=1, line=2)
boxplot(residual.sugar, col="slategray2", pch=19)
mtext("Residual Sugar", cex=0.8, side=1, line=2)
boxplot(chlorides, col="slategray2", pch=19)
mtext("Chlorides", cex=0.8, side=1, line=2)

df$color <- as.factor(df$color)
good_ones <- df$quality >= 6
bad_ones <- df$quality < 6
df[good_ones, 'quality'] <- 'good'
df[bad_ones, 'quality'] <- 'bad'
df$quality <- as.factor(df$quality)
```

```
dummies <- dummyVars(quality ~ ., data = df)
df_dummied <- data.frame(predict(dummies, newdata = df))
df_dummied[, 'quality'] <- df$quality
# set the seed for reproducibility
set.seed(1234)
trainIndices <- createDataPartition(df_dummied$quality, p = 0.7, list = FALSE)
train <- df_dummied[trainIndices, ]
test <- df_dummied[-trainIndices, ]
#1
numericColumns <- !colnames(train) %in% c('quality', 'color.red', 'color.white')
correlationMatrix <- cor(train[, numericColumns])
highlyCorrelated <- findCorrelation(correlationMatrix, cutoff = 0.6)
colnames(correlationMatrix)[highlyCorrelated]

png(paste0(today, '-', 'correlation-matrix.png'))
corrplot(correlationMatrix, method = 'number', tl.cex = 0.5)
dev.off()
#2
fitControl_rfe <- rfeControl(functions = rfFuncs, method = 'cv', number = 5) # 5-fold CV
fit_rfe <- rfe(quality ~., data = train,
           sizes = c(1:10),  # subset sizes to test (ahem... not sure how it works)
           rfeControl = fitControl_rfe)
features <- predictors(fit_rfe) # same command as fit_rfe$optVariables
max(fit_rfe$results$Accuracy)

png(paste0(today, '-', 'recursive-feature-elimination.png'))
plot(fit_rfe, type = c('g', 'o'), main = 'Recursive Feature Elimination')
dev.off()

# Normalize the quantitative variables to be within the [0,1] range
train_normalized <- preProcess(train[, numericColumns], method = 'range')
train_plot <- predict(train_normalized, train[, numericColumns])

# Let's take an initial peek at how the predictors separate on the target
png(paste0(today, '-', 'feature-plot.png'))
featurePlot(train_plot, train$quality, 'box')
dev.off()

# fitControl <- trainControl(method = 'repeatedcv', number = 5, repeats = 3)
fitControl <- trainControl(method = 'cv', number = 5)

#rf
```

```
fit_rf <- train(x = train[, features], y = train$quality,
            method = 'rf',
            # preProcess = 'range', # it seems slightly better without 'range'
            trControl = fitControl,
            tuneGrid = expand.grid(.mtry = c(2:6)),
            n.tree = 1000)
predict_rf <- predict(fit_rf, newdata = test[, features])
confusionMatrix(predict_rf, test$quality, positive = 'good')
confMat_rf <- confusionMatrix(predict_rf, test$quality, positive = 'good')
importance_rf <- varImp(fit_rf, scale = TRUE)

png(paste0(today, '-', 'importance-rf.png'))
plot(importance_rf, main = 'Feature importance for Random Forest')
dev.off()
library(xlsx)
write.xlsx(df, "K:/edwisor/CaseStudyOnWines/CaseStudyOnWines/df.xlsx")
write.xlsx(df_dummied, "K:/edwisor/CaseStudyOnWines/CaseStudyOnWines/df_dummied.xlsx")
write.xlsx(red, "K:/edwisor/CaseStudyOnWines/CaseStudyOnWines/red.xlsx")
write.xlsx(test, "K:/edwisor/CaseStudyOnWines/CaseStudyOnWines/test.xlsx")
write.xlsx(train, "K:/edwisor/CaseStudyOnWines/CaseStudyOnWines/train.xlsx")
```