\*— title: ‘Determining wine quality: A machine learning approach’ author: “Justo AndrÃ©s Manrique Urbina” date: “17 de junio de 2019” output: word\_document: default pdf\_document: default —

# Introduction

This document is the capstone project in the Data Science Professional Certificate program. As part of this program, we’ve been tasked to create our own project using a curated database. I’ve chosen a red wine quality database, which can be found in the following URL: <https://archive.ics.uci.edu/ml/datasets/Wine+Quality>. According to data description, this database is related to red variant of the Portuguese “Vinho Verde” wine. The purpose will give to this database is to predict the quality of a specific wine, given it’s variables (we’ll put more detail into this later).

This project has the following structure:

* Exploratory Data Analysis: We’ll understand the nature of our variables, the variable we want to predict and its relationship.
* Model Iteration: Given our performance metric, we’ll compare different models and see which one is a better fit. \*\* Results Discussion: We’ll determine what model is best and what are the next steps to improve our prediction.

Let’s start working!

First, let’s set up our environment. We’ll load required libraries, set up working directory and do some minimal cleansing of data (clean the headers and creating training - test partition):

# Library Load  
  
library(tidyverse)  
library(caret)  
library(e1071)  
library(corrplot)  
library(randomForest)  
  
# Environment cleaning and seeding setting  
  
rm(list=ls())  
set.seed(140)  
setwd("C:/Users/jaman/Documents/GitHub/jm-learning/data-science-cyo/data")  
  
## Data loading   
w\_red <- read\_delim("winequality-red.csv", ";", escape\_double = FALSE, trim\_ws = TRUE)  
colnames(w\_red) <- make.names(colnames(w\_red))  
  
## Creating partition  
index = createDataPartition(w\_red$quality,times=1,p=0.75, list = F)  
w\_red\_train = w\_red[index,]  
w\_red\_test = w\_red[-index,]

# Exploratory Data Analysis

* Understanding data requires some special thought on what’s the data structure and what does each data point represents. For this, we get a sense of our data structure by using formula glimple from dplyr package. After this, we understand data nature from the paper it’s originated (<http://www3.dsi.uminho.pt/pcortez/wine5.pdf>, page 23). From this understanding, we get the following ideas:
* All of our attributes are continuous.
* Not all variables have the same scale: sulfur dioxide variables are in milligrams, and the other ones are in grams.Due to this, we can see that these variables have bigger means and maximums than the other variables.
* The variable ‘quality’, which is the variable we want to predict, can be treated or as a numeric or ordinal value. This means we can do a regression or classification model. We’ll treat this variable as numeric.

glimpse(w\_red\_train)

## Observations: 1,200  
## Variables: 12  
## $ fixed.acidity <dbl> 7.4, 7.8, 7.8, 7.4, 7.4, 7.9, 7.3, 7.8, 7...  
## $ volatile.acidity <dbl> 0.700, 0.880, 0.760, 0.700, 0.660, 0.600,...  
## $ citric.acid <dbl> 0.00, 0.00, 0.04, 0.00, 0.00, 0.06, 0.00,...  
## $ residual.sugar <dbl> 1.9, 2.6, 2.3, 1.9, 1.8, 1.6, 1.2, 2.0, 6...  
## $ chlorides <dbl> 0.076, 0.098, 0.092, 0.076, 0.075, 0.069,...  
## $ free.sulfur.dioxide <dbl> 11, 25, 15, 11, 13, 15, 15, 9, 17, 17, 9,...  
## $ total.sulfur.dioxide <dbl> 34, 67, 54, 34, 40, 59, 21, 18, 102, 102,...  
## $ density <dbl> 0.9978, 0.9968, 0.9970, 0.9978, 0.9978, 0...  
## $ pH <dbl> 3.51, 3.20, 3.26, 3.51, 3.51, 3.30, 3.39,...  
## $ sulphates <dbl> 0.56, 0.68, 0.65, 0.56, 0.56, 0.46, 0.47,...  
## $ alcohol <dbl> 9.4, 9.8, 9.8, 9.4, 9.4, 9.4, 10.0, 9.5, ...  
## $ quality <dbl> 5, 5, 5, 5, 5, 5, 7, 7, 5, 5, 5, 5, 7, 5,...

summary(w\_red\_train)

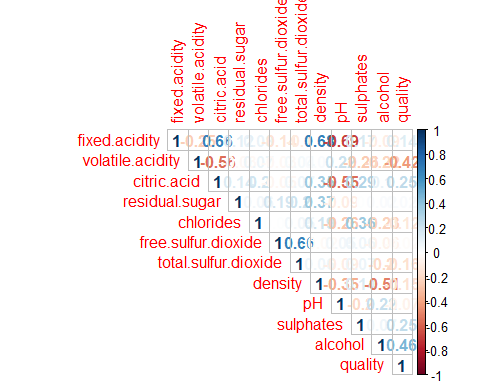
## fixed.acidity volatile.acidity citric.acid residual.sugar   
## Min. : 4.600 Min. :0.1200 Min. :0.0000 Min. : 0.900   
## 1st Qu.: 7.100 1st Qu.:0.4000 1st Qu.:0.0900 1st Qu.: 1.900   
## Median : 7.900 Median :0.5200 Median :0.2500 Median : 2.200   
## Mean : 8.309 Mean :0.5286 Mean :0.2678 Mean : 2.563   
## 3rd Qu.: 9.200 3rd Qu.:0.6400 3rd Qu.:0.4225 3rd Qu.: 2.600   
## Max. :15.600 Max. :1.5800 Max. :0.7800 Max. :15.500   
## chlorides free.sulfur.dioxide total.sulfur.dioxide  
## Min. :0.01200 Min. : 1.00 Min. : 6.00   
## 1st Qu.:0.07000 1st Qu.: 7.00 1st Qu.: 22.00   
## Median :0.07900 Median :13.00 Median : 38.00   
## Mean :0.08752 Mean :15.96 Mean : 46.70   
## 3rd Qu.:0.09100 3rd Qu.:21.00 3rd Qu.: 62.25   
## Max. :0.61100 Max. :72.00 Max. :289.00   
## density pH sulphates alcohol   
## Min. :0.9901 Min. :2.870 Min. :0.3300 Min. : 8.40   
## 1st Qu.:0.9956 1st Qu.:3.210 1st Qu.:0.5500 1st Qu.: 9.50   
## Median :0.9968 Median :3.310 Median :0.6200 Median :10.10   
## Mean :0.9968 Mean :3.311 Mean :0.6598 Mean :10.41   
## 3rd Qu.:0.9979 3rd Qu.:3.400 3rd Qu.:0.7300 3rd Qu.:11.00   
## Max. :1.0037 Max. :4.010 Max. :1.9800 Max. :14.00   
## quality   
## Min. :3.000   
## 1st Qu.:5.000   
## Median :6.000   
## Mean :5.629   
## 3rd Qu.:6.000   
## Max. :8.000

Since all of our variables are numeric, we can do an correlogram plot to understand what’s the relationship of variables. From this we gather:

* Regarding Quality variable:
* We can see alcohol and volatile acidity as possible strong predictors since they have strong correlations with quality.
* We see that sulphates and citric acid are also linearly correlatd with quality.
* We now look at the variables that are correlated with the possible predictors of quality. We see:
* Density has a inverse relationship with alcohol.

Based on this, we can use these variables for our set of predictors of quality.

m = cor(w\_red\_train[,1:12])  
corrplot(m,method="number",type="upper")



We’d like to see now if there’s a difference between the means of these predictors at each quality value. For this, we use the following summarization:

w\_red\_train %>% group\_by(quality) %>% summarise(count=n(),fsd = mean(free.sulfur.dioxide),tsd = mean(total.sulfur.dioxide),d=mean(density),ca=mean(citric.acid))

## # A tibble: 6 x 6  
## quality count fsd tsd d ca  
## <dbl> <int> <dbl> <dbl> <dbl> <dbl>  
## 1 3 8 8.25 19.2 0.997 0.0788  
## 2 4 45 12.4 38.0 0.997 0.172   
## 3 5 505 16.8 56.2 0.997 0.239   
## 4 6 479 16.1 41.7 0.997 0.271   
## 5 7 152 14.5 36.0 0.996 0.378   
## 6 8 11 12.8 32.7 0.995 0.431

From this we gather:

* Wines of higher quality has higher citric acid mean.
* Wine of the lowest quality has the lowest both citric acid mean, lowest free and total sulfure dioxide mean.
* There is small sample sizes of both low and high quality wine. We could have an unbalanced data problem.

This exploratory data analysis helped us understanding how to define our predictors.

# Model performance and selection

For this regression task, we will use the following algorithms:

* Decision Tree
* K nearest Neighbors
* Random Forest

We will evaluate this algorithms using RMSE loss function. The lower the RMSE is, the better the algorithm is. Each algorith will be cross-validated and fine tuned using caret package in R.

## Decision Tree Algorithm

Our predictors for each algorithm will be: citric acid, volatile acidity, alcohol, sulphates and density. We run the following formula for our decision tree algorithm.

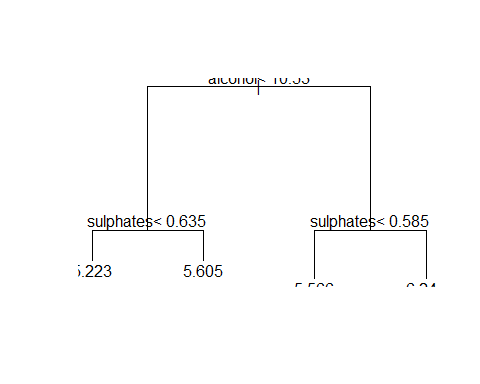
# Defining our RMSE function  
  
RMSE <- function(true\_ratings,predicted\_ratings){sqrt(mean((true\_ratings - predicted\_ratings)^2))}  
  
# defining our cross-validation technique  
  
cv = trainControl(method='repeatedcv',number = 4,repeats=2)  
  
# CART Model  
  
cartmodel = train(quality ~ citric.acid + volatile.acidity + alcohol + sulphates + density, tuneLength=4,trControl=cv, data=w\_red\_train,method='rpart')

## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info =  
## trainInfo, : There were missing values in resampled performance measures.

cart\_pr = predict(cartmodel,newdata = w\_red\_test)  
  
RMSE(cart\_pr, w\_red\_test$quality)

## [1] 0.6805127

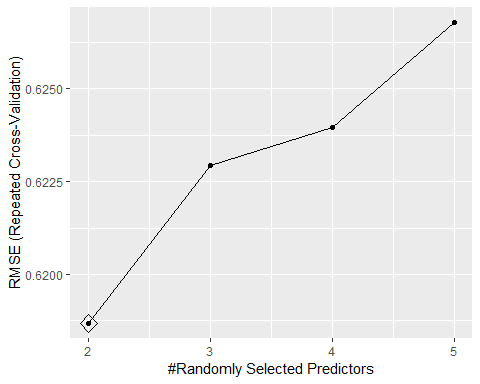
plot(cartmodel$finalModel)  
text(cartmodel$finalModel)

 We see the following results using our decision tree algorithm:

# KNN model  
  
knn = train(quality~citric.acid + volatile.acidity + alcohol + sulphates + density, w\_red\_train,method='knn')  
knn\_pr = predict(knn,newdata=w\_red\_test)  
  
RMSE(knn\_pr, w\_red\_test$quality)

## [1] 0.6436615

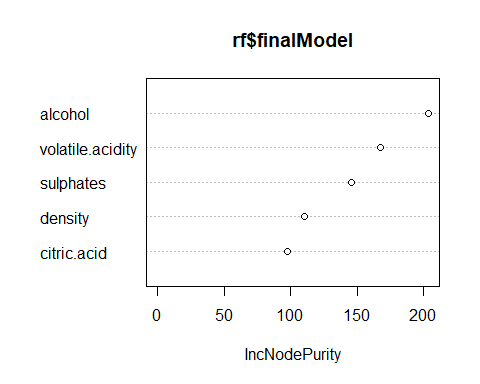
# Random Forest  
rf = train(quality~ citric.acid + volatile.acidity + alcohol + sulphates + density, w\_red\_train,method='rf',tuneLength=4,trControl=cv)  
ggplot(rf,highlight = TRUE)



rf\_pr = predict(rf, newdata = w\_red\_test)  
  
RMSE(rf\_pr, w\_red\_test$quality)

## [1] 0.5818887

## Understanding our final model  
varImpPlot(rf$finalModel)



# Results and conclusion

# Next steps