Determining wine quality: A machine learning approach

Justo Andrés Manrique Urbina

17 de junio de 2019

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)
```

```
## Warning: package 'corrplot' was built under R version 3.5.3
```

```
library(randomForest)

# Environment cleaning and seeding setting

rm(list=ls())
set.seed(140)
setwd("C:/Users/Justo.Manrique/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, 11.2, 7.4, 7.4, 7.9, 7.3, ...
## $ volatile.acidity     <dbl> 0.700, 0.880, 0.760, 0.280, 0.700, 0.660,...
## $ citric.acid          <dbl> 0.00, 0.00, 0.04, 0.56, 0.00, 0.00, 0.06,...
## $ residual.sugar       <dbl> 1.9, 2.6, 2.3, 1.9, 1.9, 1.8, 1.6, 1.2, 6...
## $ chlorides            <dbl> 0.076, 0.098, 0.092, 0.075, 0.076, 0.075,...
## $ free.sulfur.dioxide  <dbl> 11, 25, 15, 17, 11, 13, 15, 15, 17, 15, 1...
## $ total.sulfur.dioxide <dbl> 34, 67, 54, 60, 34, 40, 59, 21, 102, 65, ...
## $ density              <dbl> 0.9978, 0.9968, 0.9970, 0.9980, 0.9978, 0...
## $ pH                   <dbl> 3.51, 3.20, 3.26, 3.16, 3.51, 3.51, 3.30,...
## $ sulphates            <dbl> 0.56, 0.68, 0.65, 0.58, 0.56, 0.56, 0.46,...
## $ alcohol              <dbl> 9.4, 9.8, 9.8, 9.8, 9.4, 9.4, 9.4, 10.0, ...
## $ quality              <dbl> 5, 5, 5, 6, 5, 5, 5, 7, 5, 5, 5, 5, 5, 5,...
```

```
summary(w_red_train)
```

```
##  fixed.acidity    volatile.acidity  citric.acid     residual.sugar
##  Min.   : 4.700   Min.   :0.1200   Min.   :0.0000   Min.   : 0.900
##  1st Qu.: 7.100   1st Qu.:0.3900   1st Qu.:0.0900   1st Qu.: 1.900
##  Median : 7.900   Median :0.5200   Median :0.2600   Median : 2.200
##  Mean   : 8.333   Mean   :0.5274   Mean   :0.2701   Mean   : 2.532
##  3rd Qu.: 9.300   3rd Qu.:0.6362   3rd Qu.:0.4200   3rd Qu.: 2.600
##  Max.   :15.900   Max.   :1.5800   Max.   :0.7900   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 :14.00       Median : 37.00
##  Mean   :0.08683   Mean   :15.84       Mean   : 45.83
##  3rd Qu.:0.09000   3rd Qu.:21.00       3rd Qu.: 61.00
##  Max.   :0.61100   Max.   :72.00       Max.   :289.00
##     density             pH          sulphates         alcohol
##  Min.   :0.9901   Min.   :2.860   Min.   :0.3300   Min.   : 8.40
##  1st Qu.:0.9956   1st Qu.:3.210   1st Qu.:0.5500   1st Qu.: 9.50
##  Median :0.9967   Median :3.310   Median :0.6200   Median :10.20
##  Mean   :0.9967   Mean   :3.311   Mean   :0.6554   Mean   :10.44
##  3rd Qu.:0.9978   3rd Qu.:3.400   3rd Qu.:0.7300   3rd Qu.:11.10
##  Max.   :1.0037   Max.   :4.010   Max.   :1.9500   Max.   :14.90
##     quality
```

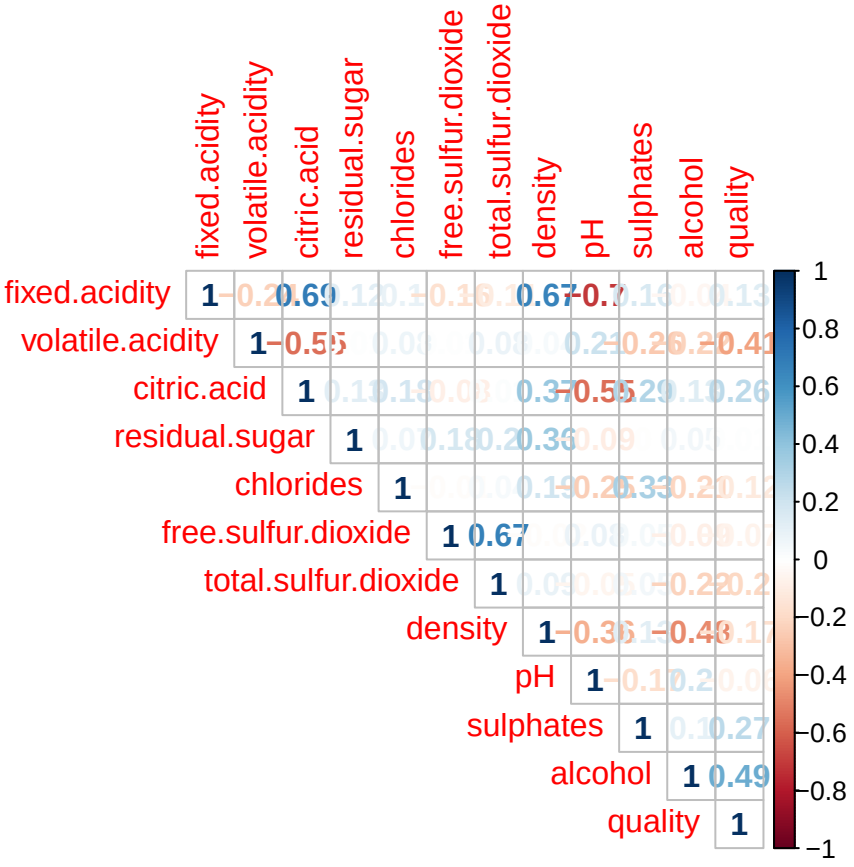
```
##  Min.   :3.000
##  1st Qu.:5.000
##  Median :6.000
##  Mean   :5.638
##  3rd Qu.:6.000
##  Max.   :8.000
```

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(to
```

```
## # A tibble: 6 x 6
##   quality count   fsd   tsd     d    ca
```

```
##     <dbl> <int> <dbl> <dbl> <dbl> <dbl>
## 1       3     8  12.2  28.2 0.997 0.209
## 2       4    38  12.2  37.4 0.997 0.143
## 3       5   512  17.1  56.0 0.997 0.236
## 4       6   479  15.6  40.1 0.997 0.279
## 5       7   148  13.6  33.6 0.996 0.384
## 6       8    15  13.3  32.9 0.995 0.385
```

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
```

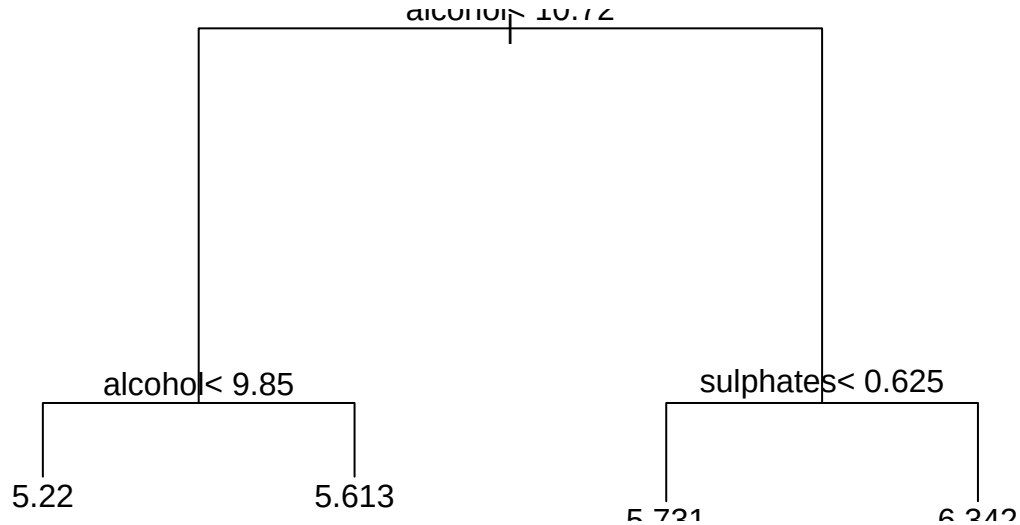
```
## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info =
## trainInfo, : There were missing values in resampled performance measures.
```

```
cart_pr = predict(cartmodel,newdata = w_red_test)

RMSE(cart_pr, w_red_test$quality)
```

```
## [1] 0.72109
```

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



We see the following results using our decision tree algorithm:

- Our RMSE is approximately 0.68.
- The predictors this algorithm has used is alcohol and sulphates. Nevertheless the range of our predicted wine quality has reduced from 3-8 to 5-6.

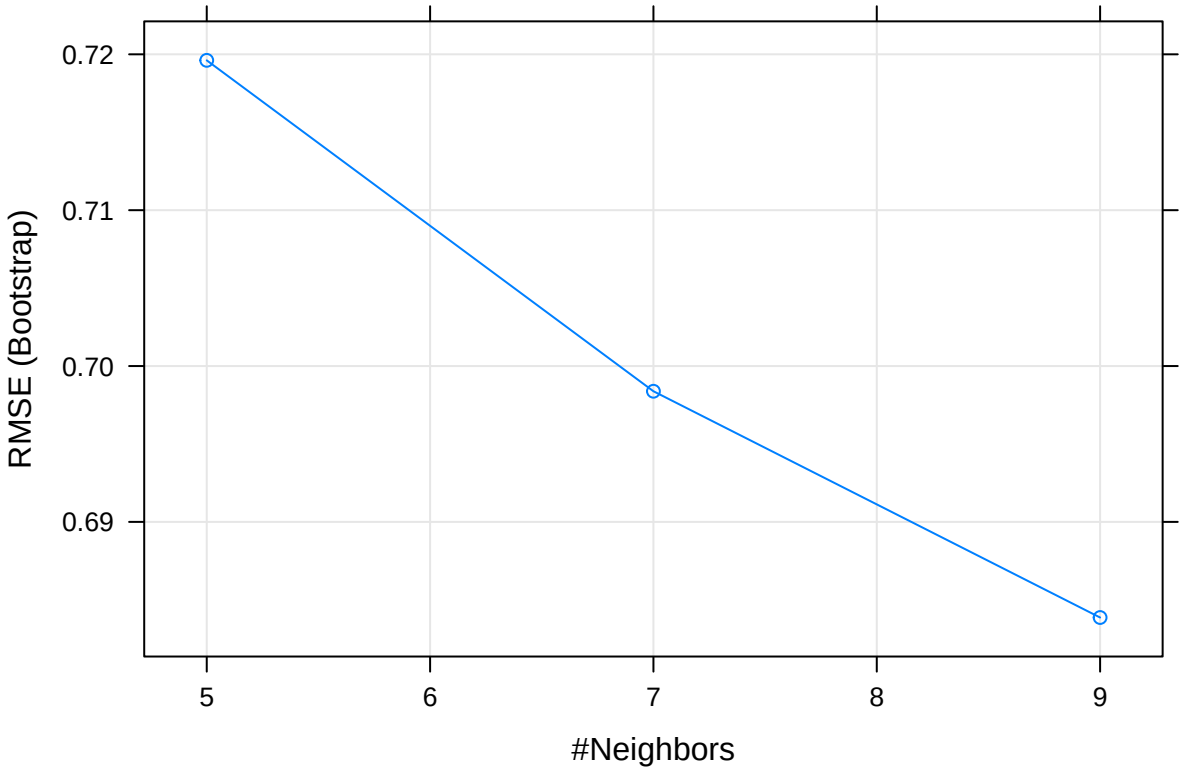
```
# KNN model

knn = train(quality~citric.acid + volatile.acidity + alcohol + sulphates + density, w_red_train,method=
knn_pr = predict(knn,newdata=w_red_test)

RMSE(knn_pr, w_red_test$quality)
```

```
## [1] 0.6561717
```

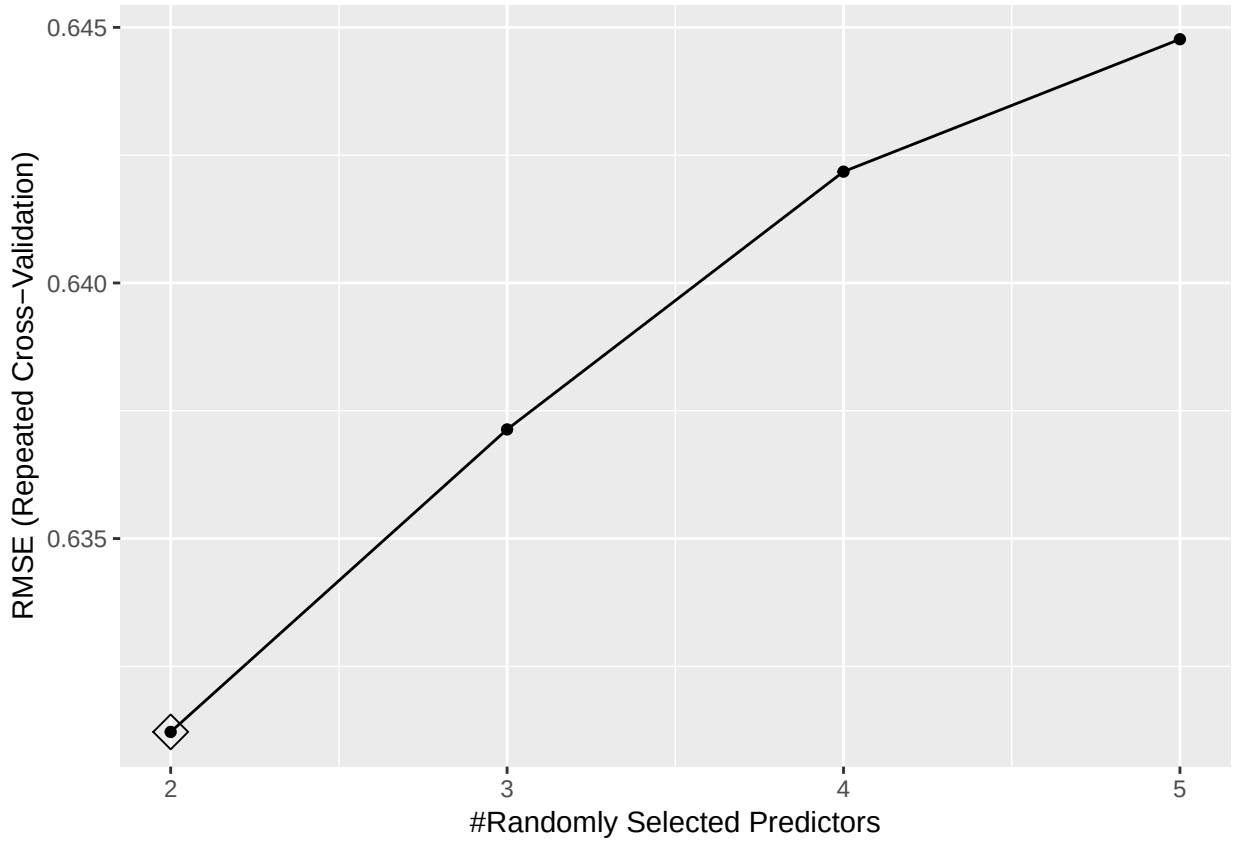
```
plot(knn)
```



We see the following results using our K nearest neighbors algorithm:

- Our RMSE has improved slightly, it's now 0.64.
- From the plot, we see that as we added more neighbors, the lower RMSE went (it's important to say that this could lead to overfitting).

```
# Random Forest
rf = train(quality~ citric.acid + volatile.acidity + alcohol + sulphates + density, w_red_train,method=
ggplot(rf,highlight = TRUE)
```

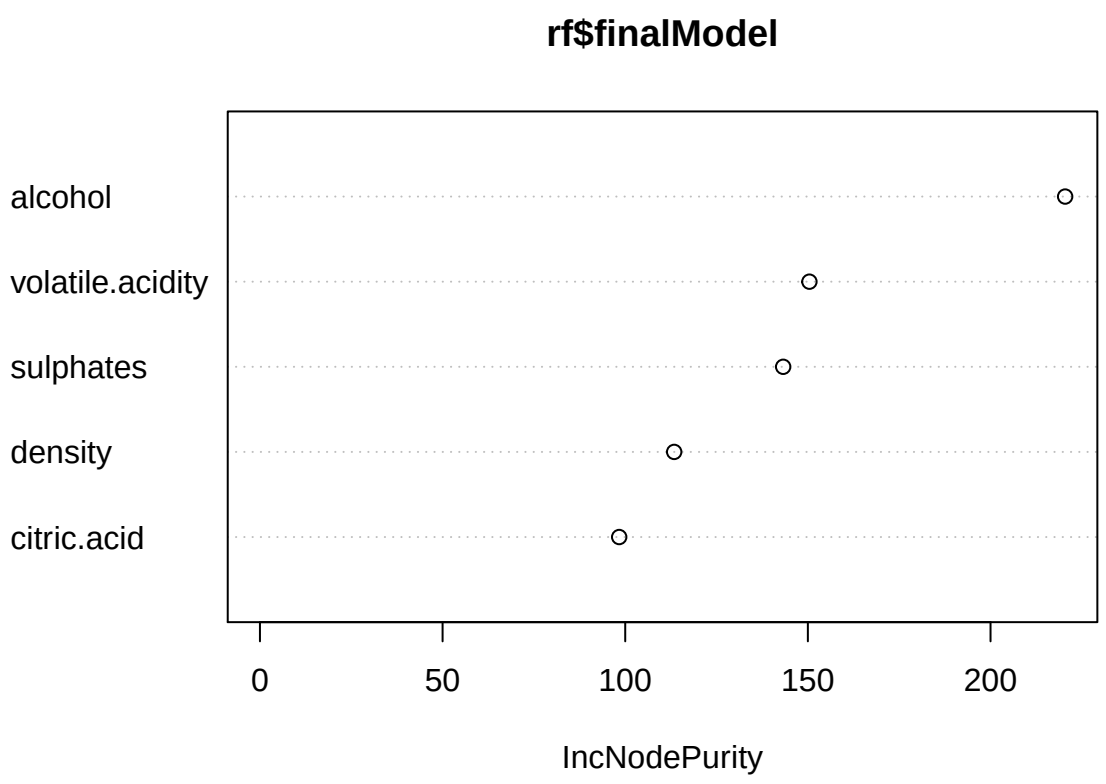



```
rf_pr = predict(rf, newdata = w_red_test)

RMSE(rf_pr, w_red_test$quality)
```

```
## [1] 0.592961
```

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



We see the following results using our random forest algorithm:

- We get a better RMSE: it's now 0.58.
- In this algorithm, alcohol and volatile acidity are the most important predictors (in decision tree's algorithm it was alcohol and sulphates).

Results and conclusion

- We applied different algorithms and evaluated its performance. From this evaluation, we've seen that the random forest algorithm performs better than the other two (0.58 vs. 0.64 and 0.68)
- Our exploratory analysis helped used defining predictors: both decision trees algorithm to random forest took alcohol and sulphates as predictors of quality.

Next steps

To further improve RMSE and get a better understanding of the data, we could augment information such as the following:

- What were the weather conditions the grape went through before being converted into wine.
- Type of production process the grape went.
- What type of nutrients the company gave to the crops.
- Have a bigger sample of low and high quality wine.