Predicting the quality of white wine

Uta Pfennig

3/9/2022

1 Introduction

This project focuses on predicting human wine preferences based on physiochemical components which were identified during wine analyses. Modeling wine preferences will generate useful insights which can be utilized to improve marketing strategies as well as wine production.

1.1 Objectives and approach

This project addresses the questions: (1) "Can wine quality be predicted by physiochemical ingredients?" and (2) "Which ingredients have the highest impact on perceived white wine quality?".

To answer these questions, the project is structured into 3 steps: * Step 1: Data exploration: Explore and visualize the data to get an overview and understand how the data is structured * Step 2: Modeling: Apply various algorithms to predict wine quality. The following models were applied: kNN model, classification tree and random forest. * Step 3: Model evaluation: Evaluate the performance of each model using the true values contained in the test set. Since not all physiochemical properties are equally important for wine quality, the variable importance for predictors will be calculated.

1.2 Data set

There are 2 data sets available from the UCI machine learning repository - one related to white wine and one for red wine. Since the project is focused on white wine. Only the data set containing white wine data has been downloaded and analysed. The data set contain 11 physicochemical and 1 sensory variables.

```
## Download file for white wine from UCI and remove temporary file
url <- "https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv"
tmp_filename <- tempfile()
download.file(url, tmp_filename)
winequality <- read.csv(tmp_filename, sep =';')
file.remove(tmp_filename)
```

```
## [1] TRUE
```

The variables below are regared as input variables based on physicochemical tests: 1 - fixed acidity 2 - volatile acidity 3 - citric acid 4 - residual sugarm 5 - chlorides 6 - free sulfur dioxide 7 - total sulfur dioxide 8 - density 9 - pH 10 - sulphates 11 - alcohol

As output variable, one variable based on sensory data is included in the data set: 12 - quality (score between 0 and 10)

Before splitting the data into train and test sets, a basic data check was conducted. The imported data set encompasses 13 variables as outlined above and contains 4898 data records.

```
## [1] 4898   12
```

All physiochecmical properties are stored as numeric values in the data frame and the quality score as integer as shown below.

```
## '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 ...
```

There are no N/A values in the data set and thus, no data cleaning related to N/A values is required.

```
sum(is.na(winequality))
```

```
## [1] 0
```

The table below lists the first 3 rows of the data set.

```
##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1           7.0             0.27        0.36           20.7     0.045
## 2           6.3             0.30        0.34            1.6     0.049
## 3           8.1             0.28        0.40            6.9     0.050
##   free.sulfur.dioxide total.sulfur.dioxide density   pH sulphates alcohol
## 1                  45                  170  1.0010 3.00      0.45     8.8
## 2                  14                  132  0.9940 3.30      0.49     9.5
## 3                  30                   97  0.9951 3.26      0.44    10.1
##   quality
## 1       6
## 2       6
## 3       6
```

2 Methods and analysis

2.1 Data exploration

In order to formulate hypothesis related to the project questions, the data set was analysed in more depth.

As starting point, a summary of each variable is provided in the overview below. The summary provides insights on the mean, median as well as the distribution of each variable.

```
##  fixed.acidity   volatile.acidity  citric.acid    residual.sugar
##  Min.   : 3.80   Min.   :0.080    Min.   :0.000   Min.   : 0.60
##  1st Qu.: 6.30   1st Qu.:0.210    1st Qu.:0.270   1st Qu.: 1.70
##  Median : 6.80   Median :0.260    Median :0.320   Median : 5.20
##  Mean   : 6.86   Mean   :0.278    Mean   :0.334   Mean   : 6.39
##  3rd Qu.: 7.30   3rd Qu.:0.320    3rd Qu.:0.390   3rd Qu.: 9.90
##  Max.   :14.20   Max.   :1.100    Max.   :1.660   Max.   :65.80
##    chlorides      free.sulfur.dioxide total.sulfur.dioxide    density
##  Min.   :0.0090   Min.   :  2.0       Min.   :  9          Min.   :0.987
##  1st Qu.:0.0360   1st Qu.: 23.0       1st Qu.:108          1st Qu.:0.992
```

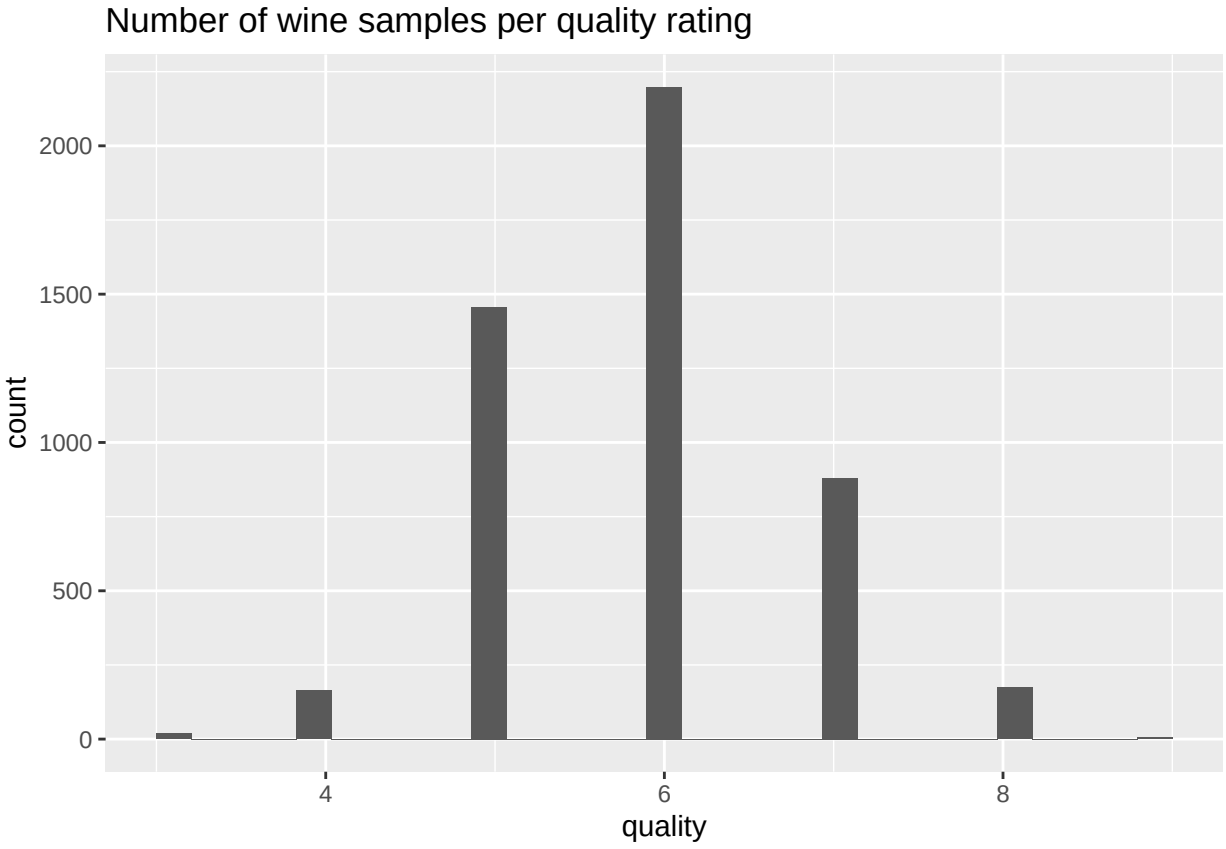
```
##  Median :0.0430   Median : 34.0       Median :134          Median :0.994
##  Mean   :0.0458   Mean   : 35.3       Mean   :138          Mean   :0.994
##  3rd Qu.:0.0500   3rd Qu.: 46.0       3rd Qu.:167          3rd Qu.:0.996
##  Max.   :0.3460   Max.   :289.0       Max.   :440          Max.   :1.039
##        pH         sulphates       alcohol        quality
##  Min.   :2.72   Min.   :0.22   Min.   : 8.0   Min.   :3.00
##  1st Qu.:3.09   1st Qu.:0.41   1st Qu.: 9.5   1st Qu.:5.00
##  Median :3.18   Median :0.47   Median :10.4   Median :6.00
##  Mean   :3.19   Mean   :0.49   Mean   :10.5   Mean   :5.88
##  3rd Qu.:3.28   3rd Qu.:0.55   3rd Qu.:11.4   3rd Qu.:6.00
##  Max.   :3.82   Max.   :1.08   Max.   :14.2   Max.   :9.00
```

The quality of the white wine was rated on a scale between 0 and 10. According to the summary chart, on average the wine quality is 5.878 (mean) with a minimum of 3 and a maximum of 9.

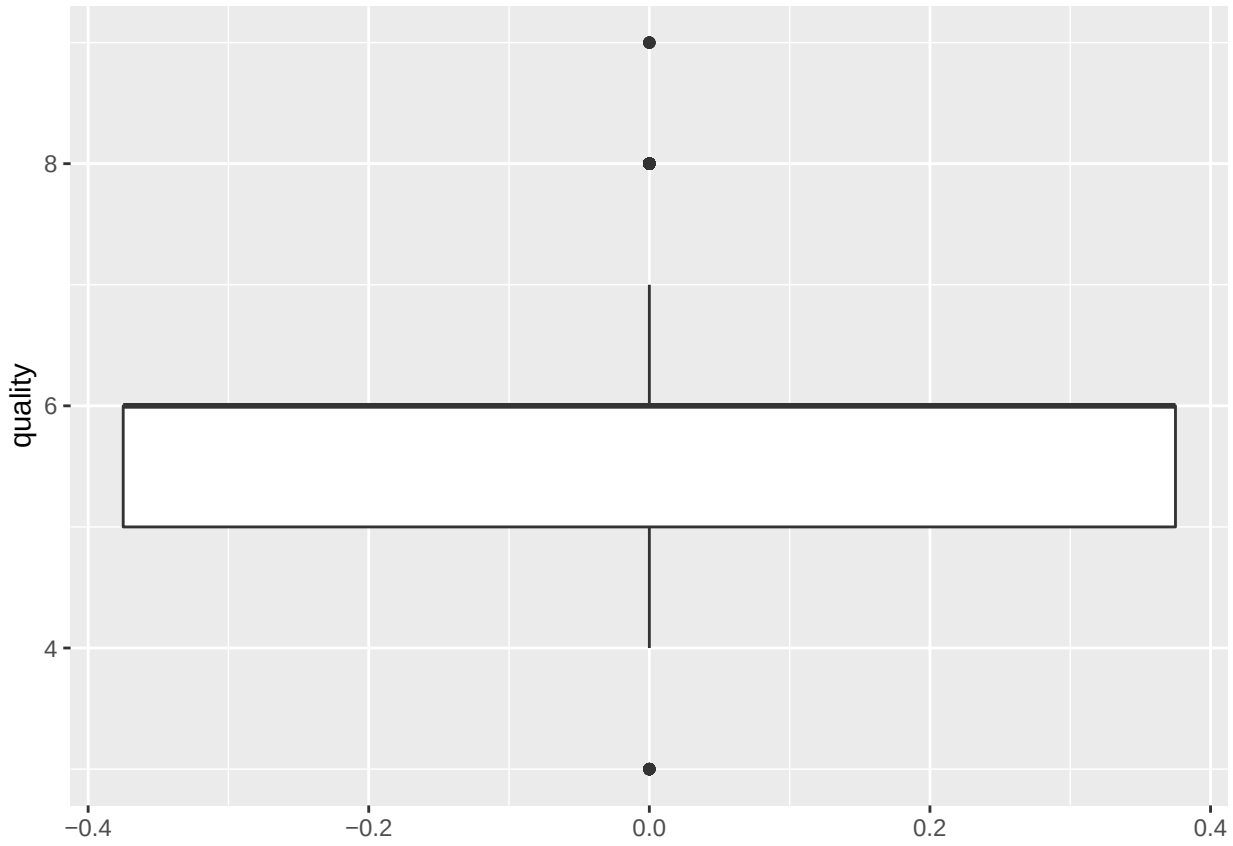
The histogram visually illustrates the distribution of wine quality across the scale.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    3.00    5.00    6.00    5.88    6.00    9.00
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



According to the boxplot, the majority of white wine samples were rated with a score of 5 or 6. There are very few data records for the quality score 3,4 and 9.



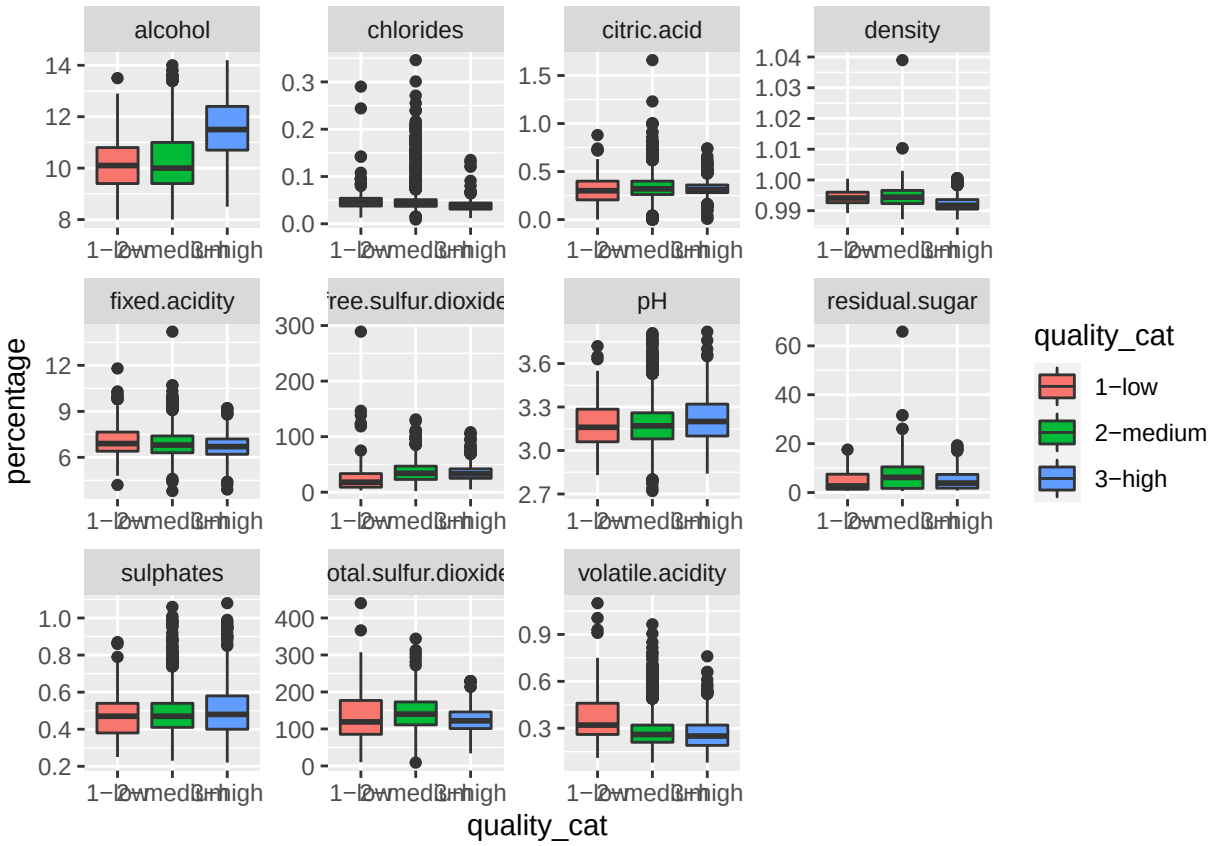
To be able to predict wine quality, the quality score will be converted into a 3-class-outcome consisting of the values "1-low" (quality <=4), "2-medium" (quality in (5,6)) and "3-high" (quality > 6).

```
winequality <- winequality %>% mutate(quality_cat = factor(case_when(
  quality %in% c(3,4) ~ "1-low",
  quality %in% c(5,6) ~ "2-medium",
  quality > 6 ~ "3-high"))) %>%
  select(-quality)
```

Is it possible to visually identify possible physiochemical properties having an impact on the wine quality and to formulate a hypothesis? To answer the question, for each ingredient the distribution per quality category is displayed in boxplots.

Based on these plots, it seems that alcohol, free sulfur dioxide as well as volatile acidity may have an impact on the wine quality. H1 - The higher the alcohol, the higher the perceived wine quality H2 - Low free sulfur dioxide result in low wine quality H3 - Low volatile acidity result in low wine quality

Different machine learning algorithms will be used in the later section to verify it.



2.2 Modeling

The downloaded and transformed data set is split into 2 data sets: (1) training data set (representing 80% of the data and used to train model) and (2) testing data set (representing 20% of the data and used to validate the model performance).

```
set.seed(1, sample.kind="Rounding")
```

```
## Warning in set.seed(1, sample.kind = "Rounding"): non-uniform 'Rounding' sampler
## used
```

```
test_index <- createDataPartition(y = winequality$quality_cat, times = 1, p = 0.2, list = FALSE)
train_set <- winequality[-test_index,]
test_set <- winequality[test_index,]
```

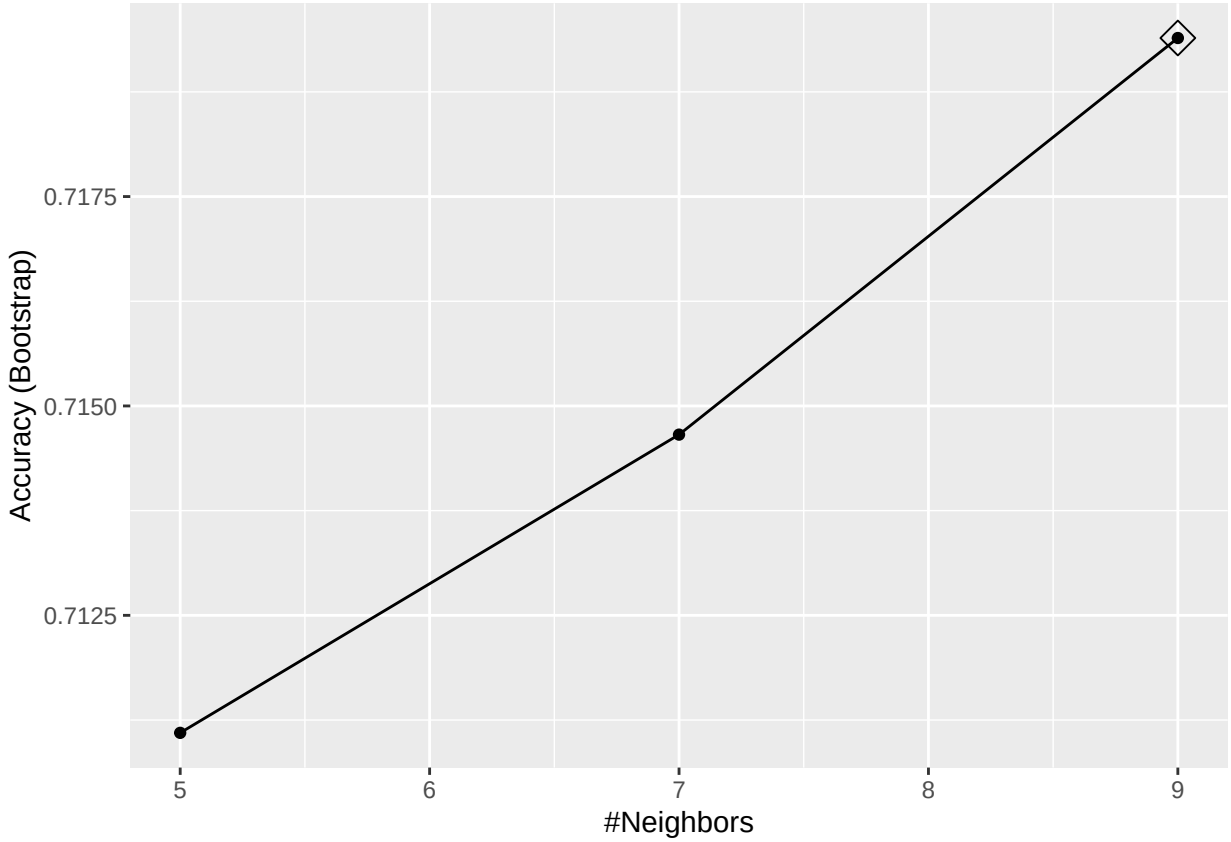
2.2.1 K-nearest Neighbor

As a starting point, wine quality was predicted using the K-nearest neighbor model. In this model the distance between each observation in the training set and each observation in the test set is computed. Wine quality has been predicted with an accuracy of 0.7388.

```
train_knn <- train(quality_cat ~ ., method = "knn", data = train_set)
y_hat_knn <- predict(train_knn, test_set, type = "raw")
accuracy_knn <- confusionMatrix(y_hat_knn, test_set$quality_cat)$overall[["Accuracy"]]
accuracy_knn
```

```
## [1] 0.7398
```

Accuracy is highest with knn = 9 as illustrated in the chart below.



The flexibility of the estimates can be controlled with the parameter k. Large ks result in smoother estimates, while smaller ks result in more flexible but wiggly estimates.

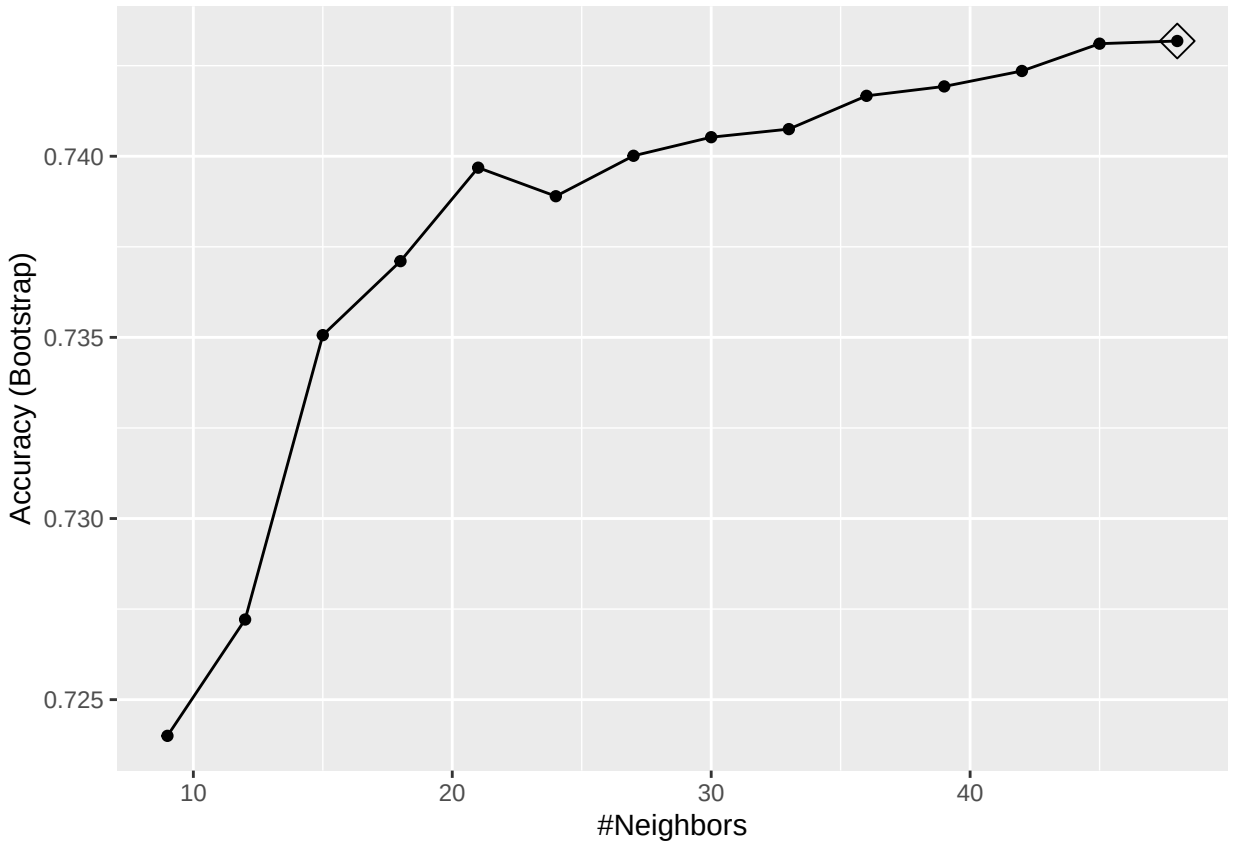
The default values of k can be changed by using the tuneGrid parameter. To optimize the kNN model for predicting wine quality, 14 different k values between 9 and 50 were included in the model. This means that 14 versions of kNN were fitted to 25 bootstrapped samples resulting in 350 kNN models.

The best performing k value is 42.

```
train_knn_2 <- train(quality_cat ~ ., method = "knn",
                     data = train_set,
                     tuneGrid = data.frame(k = seq(9, 50, 3)))
train_knn_2$bestTune
```

```
##     k
## 14 48
```

The accuracy of the cross-validation applying the different k-values (neighbor) is illustrated in the chart below.



The best performing k value is used to predict wine quality. The tuning improved slightly the accuracy of the kNN model to 0.749.

```
y_hat_knn_2 <- predict(train_knn_2, test_set, type = "raw")
accuracy_knn_2 <- confusionMatrix(y_hat_knn_2, test_set$quality_cat)$overall[["Accuracy"]]
accuracy_knn_2
```

```
## [1] 0.7449
```

method	Accuracy
Default kNN model	0.7398
Optimized kNN model (k=42)	0.7449

2.2.2 Classification and regression tree (CART)

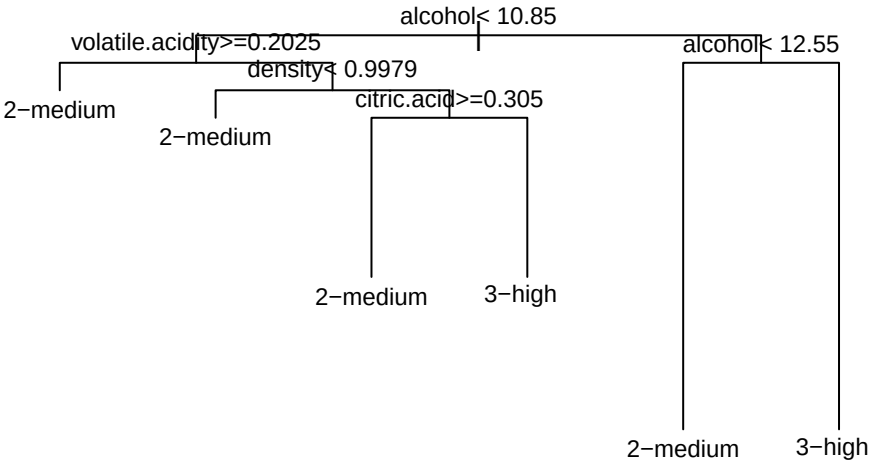
An alternative to kNN are classification trees which are used for predicting categorical outcomes. Classification trees are decision trees, which partition the data. Each node represents a test on a particular feature and each leaf represents the decision which was taken.

Classification trees are very useful and can be easily interpreted. However, the model can easily over-train. The accuracy has only slighted improved to 0.7714.

```
train_rpart <- train(quality_cat ~ ., method = "rpart", data = train_set)
y_hat_rpart <- predict(train_rpart, test_set, type = "raw")
accuracy_rpart <- confusionMatrix(y_hat_rpart, test_set$quality_cat)$overall[["Accuracy"]]
accuracy_rpart
```

```
## [1] 0.7714
```

The decision tree outlining relevant predictors are illustrated below. Alcohol and volatile acidity seem to be relevant predictors.



Alcohol, density and chlorides are the three most important variables. This means that H1 is confirmed. But there is no support for H2 and H3.

```
varImp(train_rpart)
```

```
## rpart variable importance
##
##                      Overall
## alcohol               100.00
## density                58.99
## chlorides              56.03
## total.sulfur.dioxide   19.79
## residual.sugar         18.89
## pH                     18.51
## volatile.acidity       15.87
## citric.acid            12.82
## free.sulfur.dioxide    12.27
## sulphates               4.33
## fixed.acidity           0.00
```

2.2.3 Random forest

Random forest is a very versatile machine learning algorithm which addresses the shortcomings of decision trees. By averaging multiple decision trees, the algorithm reduces instability caused by noisy data.

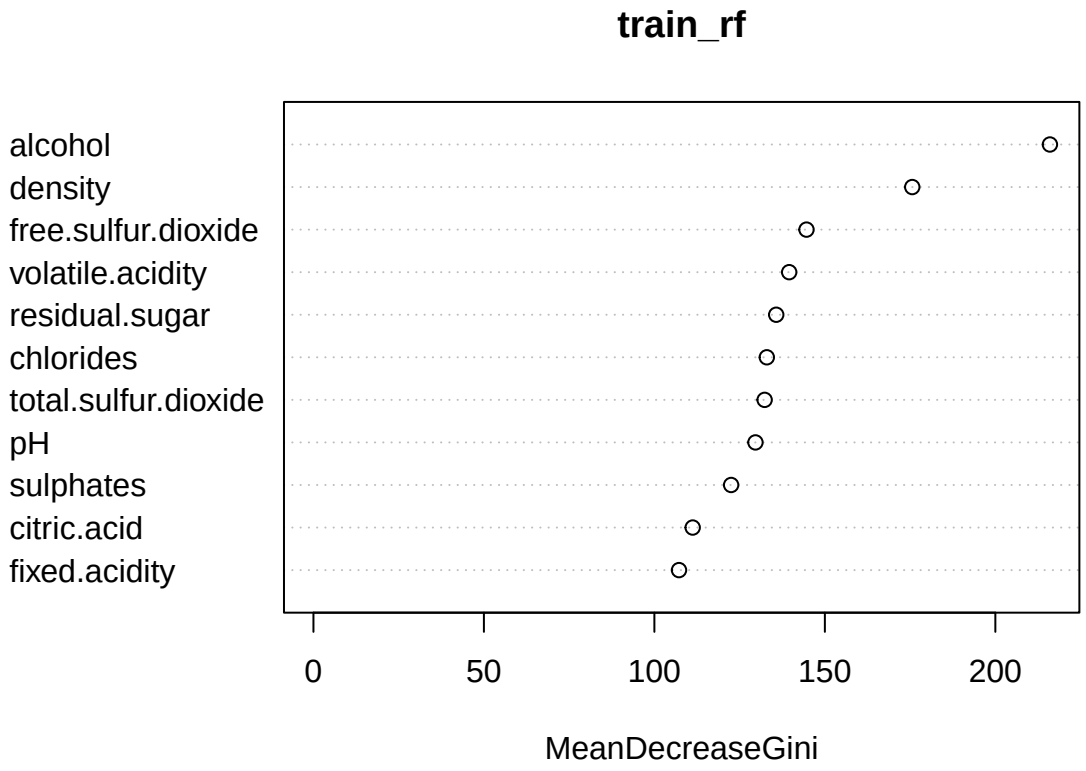
With random forest, the prediction algorithm performed best, achieving an accuracy of 0.8469.

```
train_rf <- randomForest(quality_cat ~ ., data = train_set)
accuracy_rf <- confusionMatrix(predict(train_rf, test_set),
                test_set$quality_cat)$overall["Accuracy"]
accuracy_rf
```

```
## Accuracy
##    0.851
```

Random forest reconfirmed that alcohol and density are the two most important variables. However, chlorides were listed within the top 5 variables. The least important physiochemical properties are citric acid and fixed acidity.

```
varImpPlot(train_rf)
```



3 Results

Different models have been fitted to predict white wine quality based on physiochemical properties as outlined in the results table below. Random forest performed best as prediction model.

method	Accuracy
Default kNN model	0.7398
Optimized kNN model (k=42)	0.7449
Classification tree	0.7714
Random forest	0.8510

The models confirmed that physiochemical properties can be used to predict the quality of white wine.

The exploratory data analysis suggested that alcohol, sulfur dioxide and volatile acidity might be important variables to predict wine quality. However, only alcohol was confirmed.

4 Conclusion

The results of wine quality prediction could be improved by using tuning parameters for random forest (nodesize, mtry, ntree) and by running multiple models on the training set at once (ensemble).

Outliniers were not explicitly considered in this project.

Data analysis can be further enhanced by adding further output variables.

5 References

Irizzary,R., 2018 “Introduction to Data Science”, https://rafalab.github.io/dsbook/