Data 622 - Homework 2

Leticia Salazar

October 29, 2023

Pre-work

1. Read this blog: https://decizone.com/blog/the-good-the-bad-the-ugly-of-using-decision-trees which shows some of the issues with decision trees.
2. Choose a dataset from a source in Assignment #1, or another dataset of your choice.

Assignment work

1. Based on the latest topics presented, choose a dataset of your choice and create a Decision Tree where you can solve a classification problem and predict the outcome of a particular feature or detail of the data used.
2. Switch variables* to generate 2 decision trees and compare the results. Create a random forest for regression and analyze the results.
3. Based on real cases where decision trees went wrong, and 'the bad & ugly' aspects of decision trees (https://decizone.com/blog/the-good-the-bad-the-ugly-of-using-decision-trees), how can you change this perception when using the decision tree you created to solve a real problem?

Load Libraries: Below are the libraries used to complete this assignment

```
library(tidyverse) # data prep
```

```
FALSE -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
FALSE v dplyr     1.1.3     v readr     2.1.4
FALSE v forcats   1.0.0     v stringr   1.5.0
FALSE v ggplot2   3.4.3     v tibble    3.2.1
FALSE v lubridate 1.9.3     v tidyr     1.3.0
FALSE v purrr     1.0.2
FALSE -- Conflicts --------------------------------------- tidyverse_conflicts() --
FALSE x dplyr::filter() masks stats::filter()
FALSE x dplyr::lag()    masks stats::lag()
FALSE i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become er
```

```r
library(skimr) # data prep
library(rpart) # decision tree package
library(rpart.plot) # decision tree display package
library(knitr) # kable function for table
library(tidyr) # splitting data
library(ggplot2) # graphing
library(hrbrthemes) # chart customization
```

```
FALSE NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.
FALSE       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and
FALSE       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```r
library(gridExtra) # layering charts
```

```
FALSE
FALSE Attaching package: 'gridExtra'
FALSE
FALSE The following object is masked from 'package:dplyr':
FALSE
FALSE     combine
```

```r
library(stringr) # data prep
library(tidymodels) # predictions
```

```
FALSE -- Attaching packages --------------------------------------- tidymodels 1.1.1 --
FALSE v broom        1.0.5     v rsample      1.2.0
FALSE v dials        1.2.0     v tune         1.1.2
FALSE v infer        1.0.5     v workflows    1.1.3
FALSE v modeldata    1.2.0     v workflowsets 1.0.1
FALSE v parsnip      1.1.1     v yardstick    1.2.0
FALSE v recipes      1.0.8
FALSE -- Conflicts ------------------------------------------ tidymodels_conflicts() --
FALSE x gridExtra::combine() masks dplyr::combine()
FALSE x scales::discard()    masks purrr::discard()
FALSE x dplyr::filter()      masks stats::filter()
FALSE x recipes::fixed()     masks stringr::fixed()
FALSE x dplyr::lag()         masks stats::lag()
FALSE x dials::prune()       masks rpart::prune()
FALSE x yardstick::spec()    masks readr::spec()
FALSE x recipes::step()      masks stats::step()
FALSE * Learn how to get started at https://www.tidymodels.org/start/
```

```r
library(corrplot) # correlation plot
```

```
FALSE corrplot 0.92 loaded
```

```r
library(randomForest) # for the random forest
```

```
FALSE randomForest 4.7-1.1
FALSE Type rfNews() to see new features/changes/bug fixes.
```

```
FALSE
FALSE Attaching package: 'randomForest'
FALSE
FALSE The following object is masked from 'package:gridExtra':
FALSE
FALSE     combine
FALSE
FALSE The following object is masked from 'package:dplyr':
FALSE
FALSE     combine
FALSE
FALSE The following object is masked from 'package:ggplot2':
FALSE
FALSE     margin
```

```
library(caret) # confusion matrix
```

```
FALSE Loading required package: lattice
FALSE
FALSE Attaching package: 'caret'
FALSE
FALSE The following objects are masked from 'package:yardstick':
FALSE
FALSE     precision, recall, sensitivity, specificity
FALSE
FALSE The following object is masked from 'package:purrr':
FALSE
FALSE     lift
```

Load Data: The data chosen is from Kaggle.com called Red Wine Quality. The data set is included in my GitHub and read into R.

fixed.acidity	volatile.acidity	citric.acid	residual.sugar	chlorides	free.sulfur.dioxide	total.sulfur.dioxide	density	pH	sulphates	alcohol	quality
7.4	0.70	0.00	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.8	0.88	0.00	2.6	0.098	25	67	0.9968	3.20	0.68	9.8	5
7.8	0.76	0.04	2.3	0.092	15	54	0.9970	3.26	0.65	9.8	5
11.2	0.28	0.56	1.9	0.075	17	60	0.9980	3.16	0.58	9.8	6
7.4	0.70	0.00	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.4	0.66	0.00	1.8	0.075	13	40	0.9978	3.51	0.56	9.4	5

The Data:

Based on the description from Kaggle, the two datasets are related to red and white variants of the Portuguese "Vinho Verde" wine. For more details, consult: http://www.vinhoverde.pt/en/ or the reference [Cortez et al., 2009]. Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available (e.g. there is no data about grape types, wine brand, wine selling price, etc.).

These datasets can be viewed as classification or regression tasks. The classes are ordered and not balanced (e.g. there are many more normal wines than excellent or poor ones). Outlier detection algorithms could be used to detect the few excellent or poor wines. Also, we are not sure if all input variables are relevant. So it could be interesting to test feature selection methods.

Input variables (based on physicochemical tests):

1 - fixed acidity

2 - volatile acidity

3 - citric acid

4 - residual sugar

5 - chlorides

6 - free sulfur dioxide

7 - total sulfur dioxide

8 - density

9 - pH

10 - sulphates

11 - alcohol

Output variable (based on sensory data):

12 - quality (score between 0 and 10)

Data Exploration:

Using the `skimr` library we can obtain a quick summary statistic of the dataset. It has 1599 values with 12 variables all numeric and no missing variables.

Table 2: Data summary

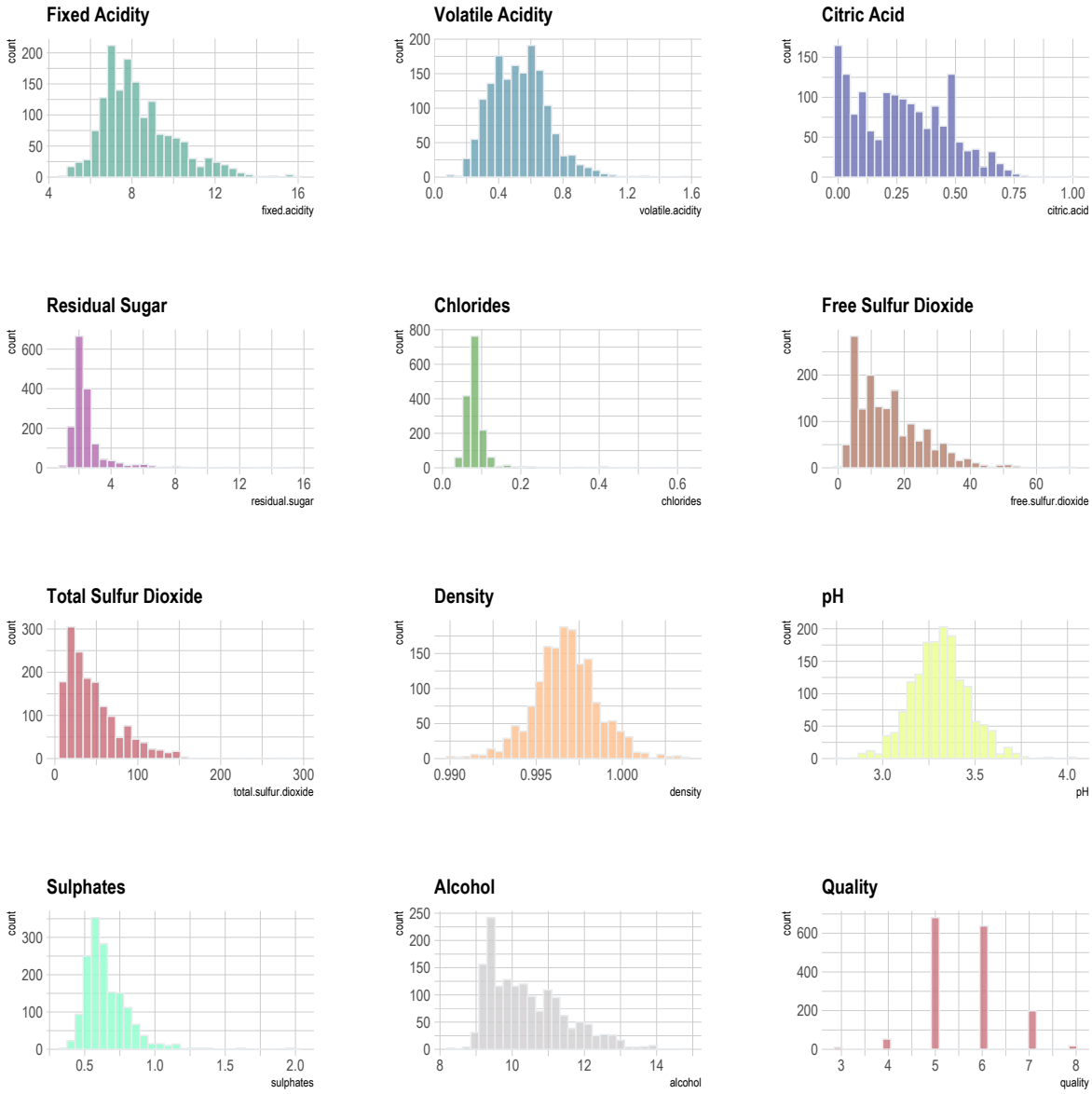
Name	wine_df
Number of rows	1599
Number of columns	12
Column type frequency:	
numeric	12
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
fixed.acidity	0	1	8.32	1.74	4.60	7.10	7.90	9.20	15.90	
volatile.acidity	0	1	0.53	0.18	0.12	0.39	0.52	0.64	1.58	
citric.acid	0	1	0.27	0.19	0.00	0.09	0.26	0.42	1.00	
residual.sugar	0	1	2.54	1.41	0.90	1.90	2.20	2.60	15.50	
chlorides	0	1	0.09	0.05	0.01	0.07	0.08	0.09	0.61	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
free.sulfur.dioxide	0	1	15.87	10.46	1.00	7.00	14.00	21.00	72.00	
total.sulfur.dioxide	0	1	46.47	32.90	6.00	22.00	38.00	62.00	289.00	
density	0	1	1.00	0.00	0.99	1.00	1.00	1.00	1.00	
pH	0	1	3.31	0.15	2.74	3.21	3.31	3.40	4.01	
sulphates	0	1	0.66	0.17	0.33	0.55	0.62	0.73	2.00	
alcohol	0	1	10.42	1.07	8.40	9.50	10.20	11.10	14.90	
quality	0	1	5.64	0.81	3.00	5.00	6.00	6.00	8.00	

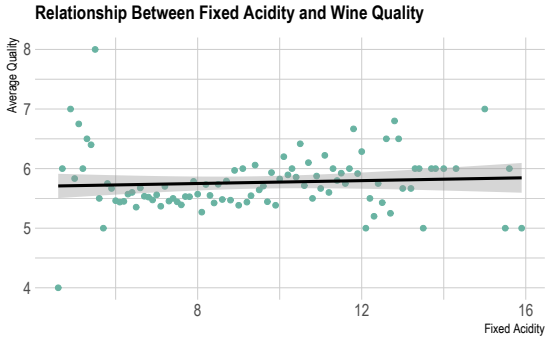
Let's take a look at the distributions of the data set:

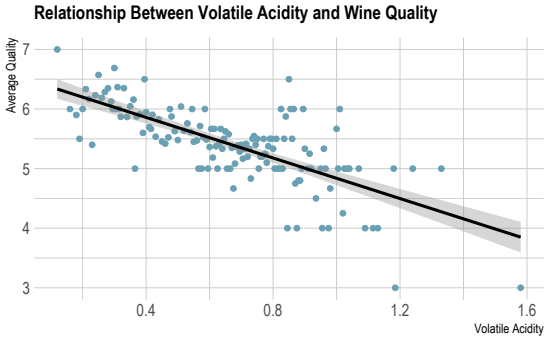


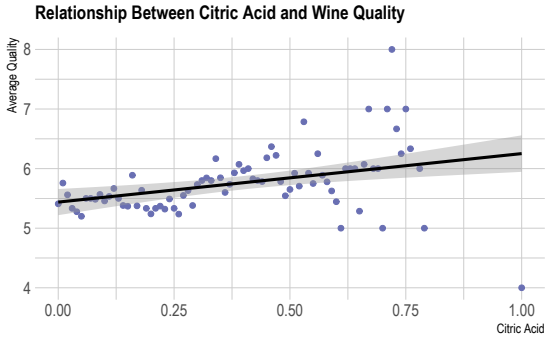
Some notes on the visualizations above:

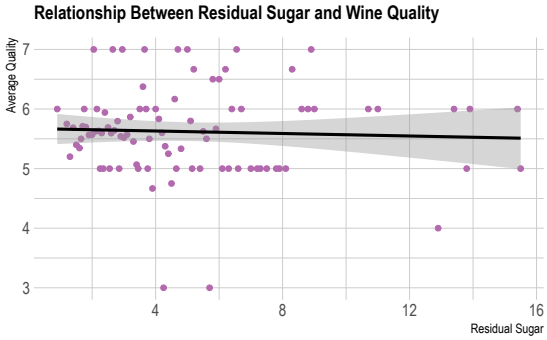
- Most of the distributions for the variables are right skewed with the exception of Density and pH
- Density and pH have more of a normal distribution
- Citric Acid has a more uniform distribution

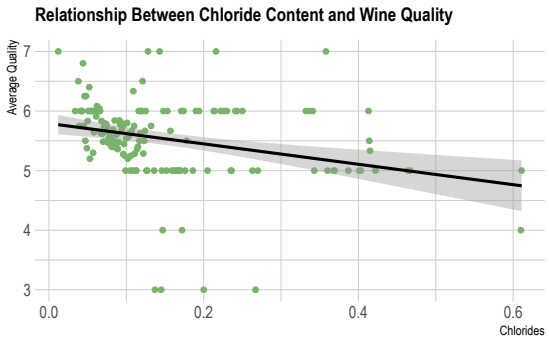
Let's check if there's any relationships between the variables against the quality of the wine:

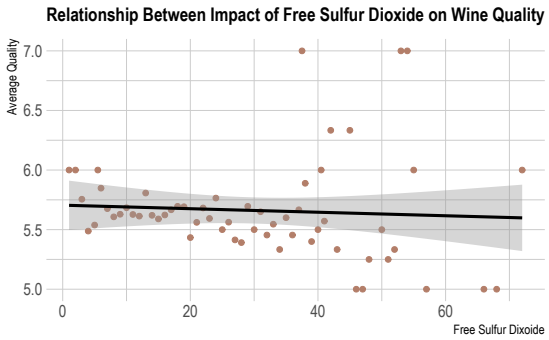


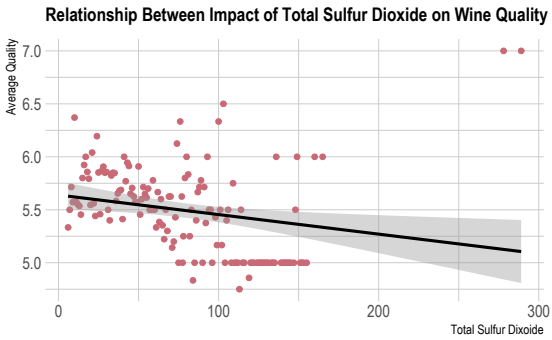


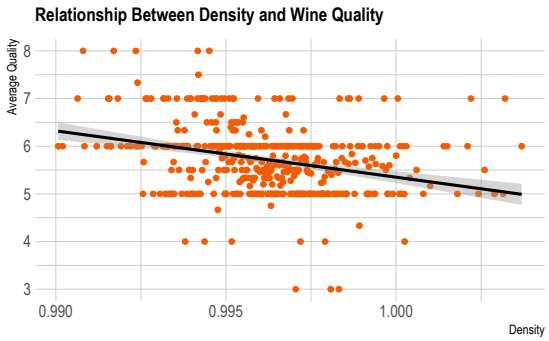


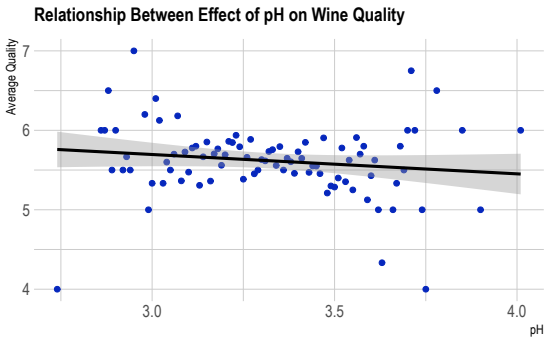


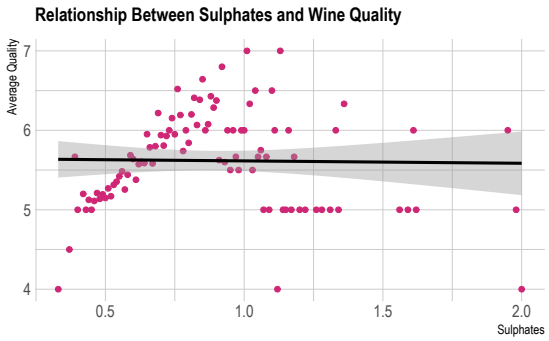


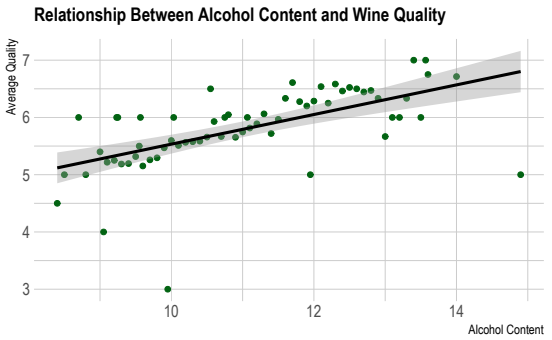












Key takeaways from the scatterplot:

- There is no correlation between a wine's residual sugar and its quality rating.
- There's no visible relationship between chloride content, free sulfur dioxide, and wine quality.
- Wines containing higher levels of total sulfur dioxide are not consistently rated as low quality wines and don't provide a reliable indicator of wine quality.
- There is a slight negative relationship between a wine's density and it's quality rating. Higher density wines tend to have a slightly lower quality rating.
- There is very little to no correlation between pH and wine quality.

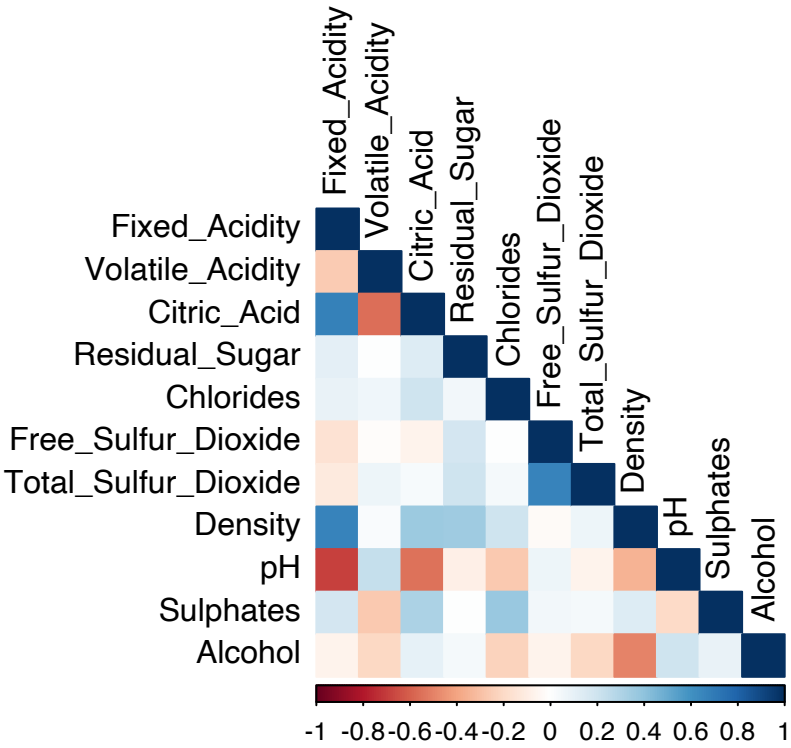
- There is a slight positive relationship between alcohol content and wine quality. The higher the alcohol content, the higher the average of the wine quality.

Data Preparation:

Now that I've visualized the data I want to do one minor change to the columns. Most of the columns have a "." and I'm changing it to an "_". Since there's no missing values, and all values are already numeric, there's not much to prepare the data.

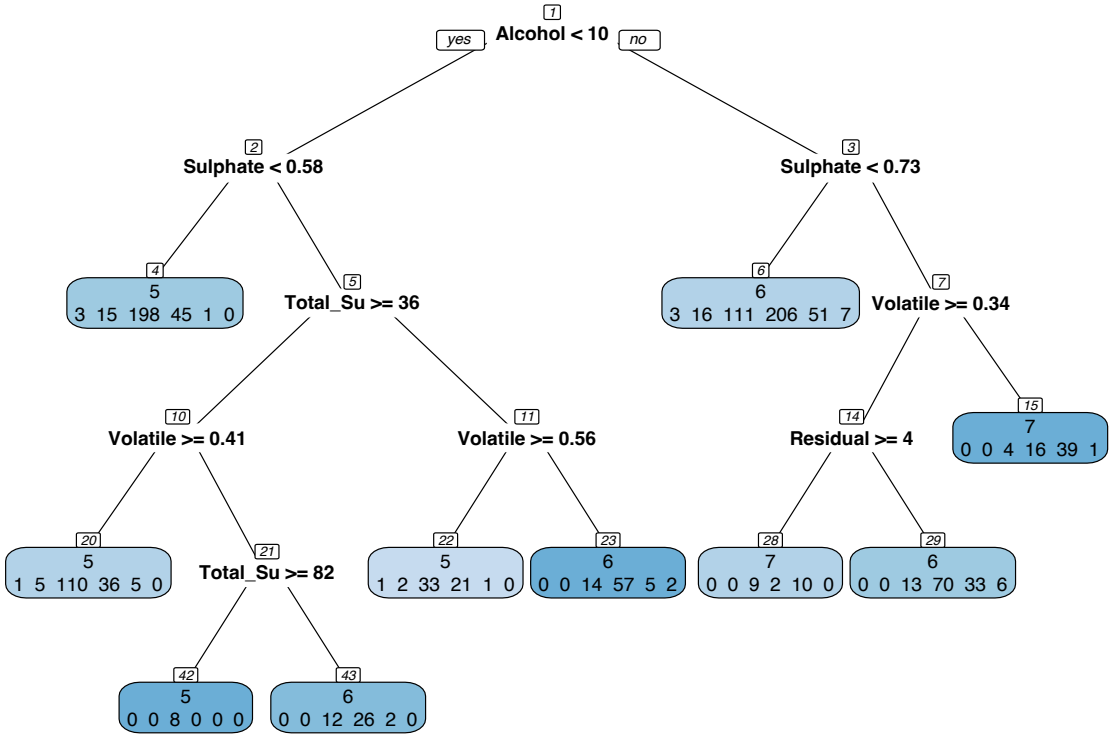
Fixed_Acidity	Volatile_Acidity	Citric_Acid	Residual_Sugar	Chlorides	Free_Sulfur_Dioxide	Total_Sulfur_Dioxide	Density	pH	Sulphates	Alcohol	Quality
7.4	0.70	0.00	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.8	0.88	0.00	2.6	0.098	25	67	0.9968	3.20	0.68	9.8	5
7.8	0.76	0.04	2.3	0.092	15	54	0.9970	3.26	0.65	9.8	5
11.2	0.28	0.56	1.9	0.075	17	60	0.9980	3.16	0.58	9.8	6
7.4	0.70	0.00	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5
7.4	0.66	0.00	1.8	0.075	13	40	0.9978	3.51	0.56	9.4	5

The correlation plot below is measuring the degree of linear relationship within the dataset. The values in which this is measured falls between -1 and +1, with +1 being a strong positive correlation and -1 a strong negative correlation. The darker the dot the more strongly correlated (whether positive or negative). From the results below, there's a strong positive correlation with citric acid, density and fixed acidity as well as free sulfur dioxide and total sulfur dioxide. Negative strong correlations are only seen with fixed acidity and pH, citric acid and volatile acidy, citric acid and pH, and density and alcohol.



Model Building:

We have to create two decision tree models and one random forest model. The first decision tree is between `Quality` and the whole data set. I started off by doing the cross validations setup by using the 75:25 ratio. After that we then created the decision tree seen below:



Then we test the model using the validation dataset to create the prediction table below:

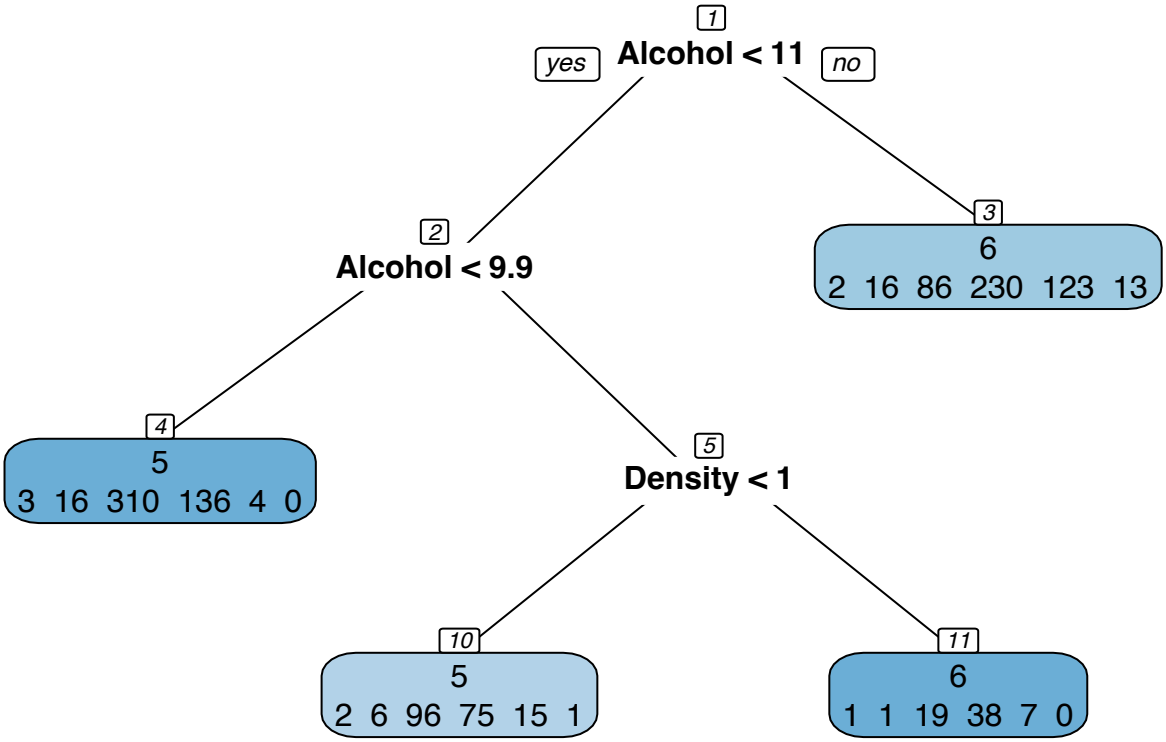
	3	4	5	6	7	8
3	0	0	1	1	0	0
4	0	0	7	8	0	0
5	0	0	110	55	4	0
6	0	0	42	106	11	0
7	0	0	2	37	13	0
8	0	0	0	1	1	0

and we check the accuracy which is 57.4%:

Table 6: Accuracy

x
0.5739348

Switching Variables: For the second decision tree I will be looking at the relationship between `Quality` and `Density`, `pH`, and `Alcohol`. I created a new dataset from the original choosing only the variables above. Following the same step to create the first decision tree, we create the second:



Same as before, we create the prediciton table:

	3	4	5	6	7	8
3	0	0	1	1	0	0
4	0	0	11	3	0	0
5	0	0	138	32	0	0
6	0	0	70	89	0	0
7	0	0	9	41	0	0
8	0	0	1	3	0	0

and now for the accuracy of 56.8% which is lower than the first decision tree:

Table 8: Accuracy

x
0.5689223

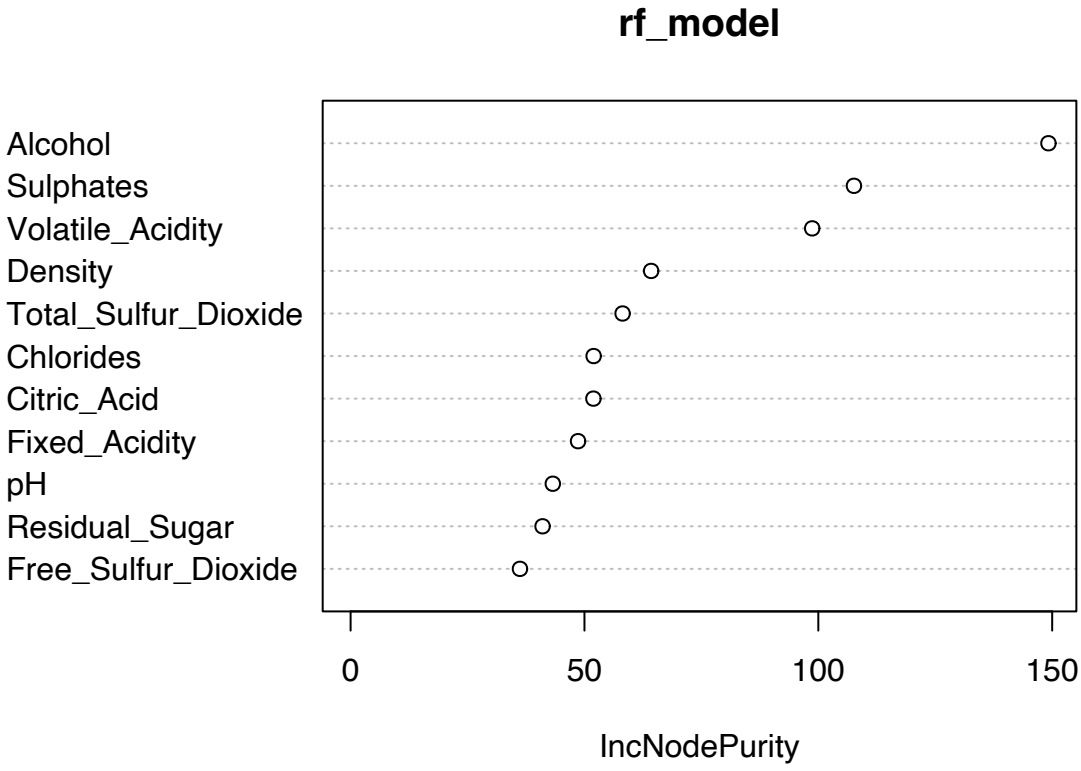
Random Forest A Random Forest is an ensemble learning technique in machine learning that combines multiple decision trees to make accurate predictions. It works by creating a collection of decision trees, each trained on a bootstrapped dataset (randomly sampled with replacement) from the original data and considering only a subset of features at each split. The final prediction in a classification task is determined

by a majority vote of the individual trees, while in a regression task, it's an average of their predictions. Random Forests are valued for their high accuracy, resistance to overfitting, and the ability to assess feature importance.

For the random forest model, I am choosing the first decision tree as it had a higher accuracy compared to the second model. First we create the random forest model using the training data and then applying it to the validation data.

```
##
## Call:
##  randomForest(formula = Quality ~ ., data = wine_train)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
##           Mean of squared residuals: 0.340287
##                     % Var explained: 48.36
```

From the random forest model we created, we can create a variable importance plot which shows each variable and how important it is in classifying the data. From the plot below we note that `Alcohol`, `Sulphates` and `Volatile_Acidity` are among the top variables that play a significant role in the classification of the quality of the wine.



Numerically, we can see the same result below:

	Overall
Fixed_Acidity	48.62991
Volatile_Acidity	98.69333
Citric_Acid	51.91002
Residual_Sugar	41.04249
Chlorides	51.96039
Free_Sulfur_Dioxide	36.21257
Total_Sulfur_Dioxide	58.15330
Density	64.25719
pH	43.23137
Sulphates	107.61902
Alcohol	149.20112

Lastly, I perform the random forest on the validation data to check the accuracy of the model with the results below:

```
# create some random number for reproduction
set.seed(4)

# creating random forest model using the validation data
rf_pred <- predict(rf_model,newdata = wine_valid)

# confusion matrix output
#confusionMatrix(rf_pred, wine_valid$Quality)
```

Conclusion:

To change the perception of decision trees, especially considering their limitations and instances where they've gone wrong, you can adopt various strategies when using a decision tree to address real problems. Acknowledge their limitations and be transparent about what they can and cannot do. Focus on data quality and preprocessing to ensure the best input. Implement techniques to control overfitting, such as pruning or ensembling. Choose relevant features and maintain interpretability, explaining the tree's decisions transparently. Continuously monitor and update the model, document the process, and conduct sensitivity analyses. Additionally, consider ethical aspects and educate stakeholders on the strengths and weaknesses of decision trees, ultimately promoting a more informed and realistic perspective on their utility. However, like any tool, they can have limitations and drawbacks. In this homework my set-up to fully completing the random forest was an error that read "Error: `data` and `reference` should be factors with the same levels." which I hope to be able to correct.