# Chapter_6_regression_trees_example

Fran Camacho

2025-02-11

## Chapter 6 - Regression trees example

Example from the book "Machine Learning with R", by Brett Lantz:

Estimating the quality of wines with regression trees and model trees

### Step 1 – collecting data

In this case study, we will use regression trees and model trees to create a system capable of mimicking expert ratings of wine. Because trees result in a model that is readily understood, this could allow winemakers to identify key factors that contribute to better-rated wines.

### Step 2 – exploring and preparing the data

```r
# install needed packages
#install.packages("rpart")  # rpart (recursive partitioning) package offers the
                            # most faithful implementation of regression trees
library(rpart)

#install.packages("rpart.plot")
library(rpart.plot)
```

```r
# import the CSV file
wine <- read.csv(file.path("Chapter06", "whitewines.csv"), stringsAsFactors = TRUE)
```

The wine data includes 11 features and the quality outcome, as follows:

```r
str(wine)
```

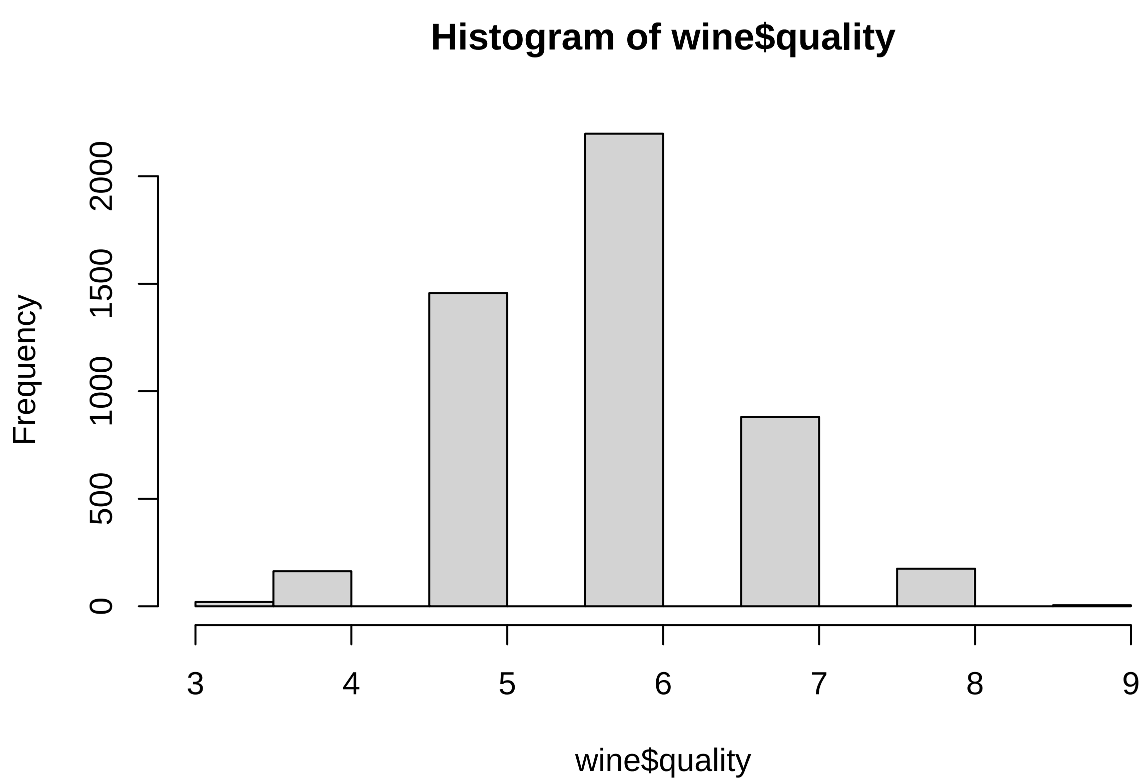
```
## 'data.frame':    4898 obs. of  12 variables:
##  $ fixed.acidity       : num  6.7 5.7 5.9 5.3 6.4 7 7.9 6.6 7 6.5 ...
##  $ volatile.acidity    : num  0.62 0.22 0.19 0.47 0.29 0.14 0.12 0.38 0.16 0.37 ...
##  $ citric.acid         : num  0.24 0.2 0.26 0.1 0.21 0.41 0.49 0.28 0.3 0.33 ...
##  $ residual.sugar      : num  1.1 16 7.4 1.3 9.65 0.9 5.2 2.8 2.6 3.9 ...
##  $ chlorides           : num  0.039 0.044 0.034 0.036 0.041 0.037 0.049 0.043 0.043 0.027 ...
##  $ free.sulfur.dioxide : num  6 41 33 11 36 22 33 17 34 40 ...
##  $ total.sulfur.dioxide: num  62 113 123 74 119 95 152 67 90 130 ...
```

```
##  $ density              : num  0.993 0.999 0.995 0.991 0.993 ...
##  $ pH                   : num  3.41 3.22 3.49 3.48 2.99 3.25 3.18 3.21 2.88 3.28 ...
##  $ sulphates            : num  0.32 0.46 0.42 0.54 0.34 0.43 0.47 0.47 0.47 0.39 ...
##  $ alcohol              : num  10.4 8.9 10.1 11.2 10.9 ...
##  $ quality              : int  5 6 6 4 6 6 6 6 6 7 ...
```

Compared with other types of machine learning models, one of the advantages of trees is that they can handle many types of data without preprocessing. This means we do not need to normalize or standardize the features.

However, a bit of effort to examine the distribution of the outcome variable is needed to inform our evaluation of the model's performance. For instance, suppose that there was very little vari- ation in quality from wine to wine, or that wines fell into a bimodal distribution: either very good or very bad. This may impact the way we design the model. To check for such extremes, we can examine the distribution of wine quality using a histogram:

```
hist(wine$quality)
```




The wine quality values appear to follow a roughly normal, bell-shaped distribution, centered around a value of six. This makes sense intuitively, because most wines are of average quality.

Our last step, then, is to divide the dataset into training and testing sets. Since the wine dataset was already sorted randomly, we can partition it into two sets of contiguous rows as follows:

```
wine_train <- wine[1:3750, ]
wine_test  <- wine[3751:4898, ]
```

## Step 3 – training a model on the data

```r
#resulting regression tree model object is named m.rpart to distinguish it from the model tree we will
m.rpart <- rpart(quality ~ ., data = wine_train)
```

For basic information about the tree, simply type the name of the model object:

```r
m.rpart
```

```
## n= 3750
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 3750 2945.53200 5.870933
##    2) alcohol< 10.85 2372 1418.86100 5.604975
##      4) volatile.acidity>=0.2275 1611  821.30730 5.432030
##        8) volatile.acidity>=0.3025 688  278.97670 5.255814 *
##        9) volatile.acidity< 0.3025 923  505.04230 5.563380 *
##      5) volatile.acidity< 0.2275 761  447.36400 5.971091 *
##    3) alcohol>=10.85 1378 1070.08200 6.328737
##      6) free.sulfur.dioxide< 10.5 84   95.55952 5.369048 *
##      7) free.sulfur.dioxide>=10.5 1294  892.13600 6.391036
##       14) alcohol< 11.76667 629  430.11130 6.173291
##         28) volatile.acidity>=0.465 11   10.72727 4.545455 *
##         29) volatile.acidity< 0.465 618  389.71680 6.202265 *
##       15) alcohol>=11.76667 665  403.99400 6.596992 *
```

For each node in the tree, the number of examples reaching the decision point is listed. For instance, all 3,750 examples begin at the root node, of which 2,372 have alcohol $<$ 10.85 and 1,378 have alcohol $>=$ 10.85.

Nodes indicated by * are terminal or leaf nodes, which means that they result in a prediction (listed here as yval). For example, node 5 has a yval of 5.971091. When the tree is used for predictions, any wine samples with alcohol $<$ 10.85 and volatile.acidity $<$ 0.2275 would therefore be predicted to have a quality value of 5.97.

A more detailed summary of the tree's fit:

```r
summary(m.rpart)
```

```
## Call:
## rpart(formula = quality ~ ., data = wine_train)
##   n= 3750
##
##           CP nsplit rel error    xerror       xstd
## 1 0.15501053      0 1.0000000 1.0002107 0.02445665
## 2 0.05098911      1 0.8449895 0.8456859 0.02335357
## 3 0.02796998      2 0.7940004 0.8025987 0.02276633
## 4 0.01970128      3 0.7660304 0.7779622 0.02156654
## 5 0.01265926      4 0.7463291 0.7584318 0.02078557
## 6 0.01007193      5 0.7336698 0.7517638 0.02067768
## 7 0.01000000      6 0.7235979 0.7456693 0.02065742
```

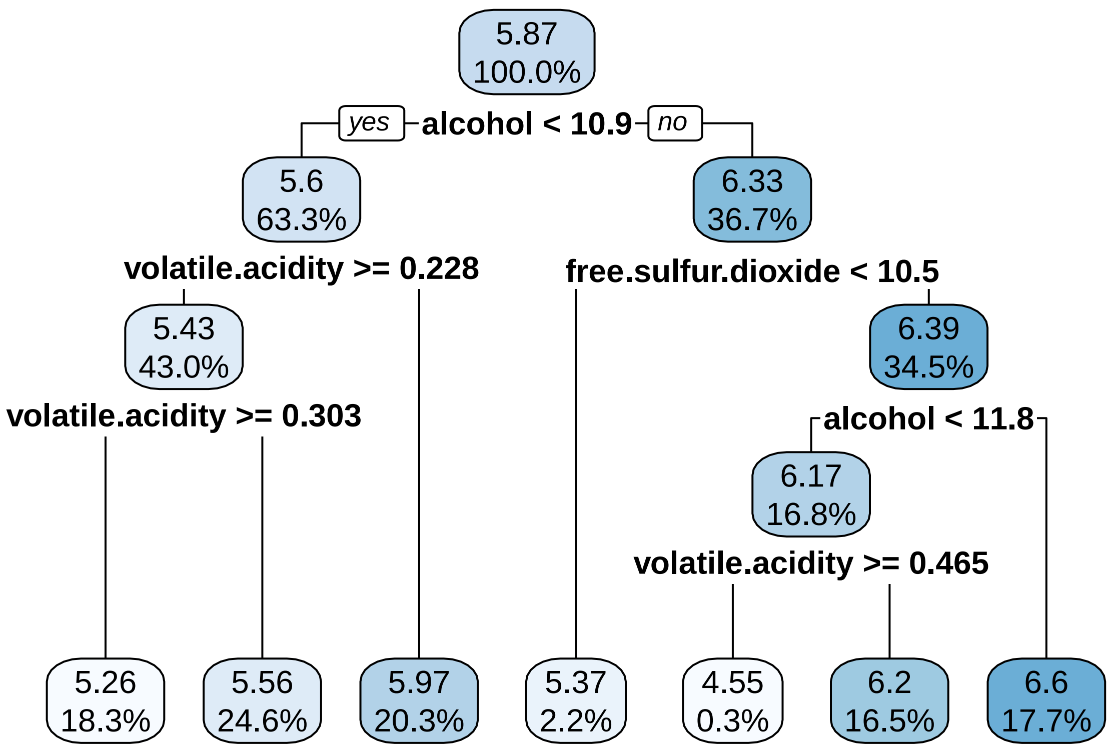
```
##
## Variable importance
##              alcohol              density     volatile.acidity
##                   34                   21                   15
##            chlorides total.sulfur.dioxide  free.sulfur.dioxide
##                   11                    7                    6
##       residual.sugar            sulphates          citric.acid
##                    3                    1                    1
##
## Node number 1: 3750 observations,    complexity param=0.1550105
##   mean=5.870933, MSE=0.7854751
##   left son=2 (2372 obs) right son=3 (1378 obs)
##   Primary splits:
##       alcohol              < 10.85    to the left,  improve=0.15501050, (0 missing)
##       density              < 0.992035 to the right, improve=0.10915940, (0 missing)
##       chlorides            < 0.0395   to the right, improve=0.07682258, (0 missing)
##       total.sulfur.dioxide < 158.5    to the right, improve=0.04089663, (0 missing)
##       citric.acid          < 0.235    to the left,  improve=0.03636458, (0 missing)
##   Surrogate splits:
##       density              < 0.991995 to the right, agree=0.869, adj=0.644, (0 split)
##       chlorides            < 0.0375   to the right, agree=0.757, adj=0.339, (0 split)
##       total.sulfur.dioxide < 103.5    to the right, agree=0.690, adj=0.155, (0 split)
##       residual.sugar       < 5.375    to the right, agree=0.667, adj=0.094, (0 split)
##       sulphates            < 0.345    to the right, agree=0.647, adj=0.038, (0 split)
##
## Node number 2: 2372 observations,    complexity param=0.05098911
##   mean=5.604975, MSE=0.5981709
##   left son=4 (1611 obs) right son=5 (761 obs)
##   Primary splits:
##       volatile.acidity    < 0.2275   to the right, improve=0.10585250, (0 missing)
##       free.sulfur.dioxide < 13.5     to the left,  improve=0.03390500, (0 missing)
##       citric.acid         < 0.235    to the left,  improve=0.03204075, (0 missing)
##       alcohol             < 10.11667 to the left,  improve=0.03136524, (0 missing)
##       chlorides           < 0.0585   to the right, improve=0.01633599, (0 missing)
##   Surrogate splits:
##       pH                   < 3.485    to the left,  agree=0.694, adj=0.047, (0 split)
##       sulphates            < 0.755    to the left,  agree=0.685, adj=0.020, (0 split)
##       total.sulfur.dioxide < 105.5    to the right, agree=0.683, adj=0.011, (0 split)
##       residual.sugar       < 0.75     to the right, agree=0.681, adj=0.007, (0 split)
##       chlorides            < 0.0285   to the right, agree=0.680, adj=0.003, (0 split)
##
## Node number 3: 1378 observations,    complexity param=0.02796998
##   mean=6.328737, MSE=0.7765472
##   left son=6 (84 obs) right son=7 (1294 obs)
##   Primary splits:
##       free.sulfur.dioxide  < 10.5     to the left,  improve=0.07699080, (0 missing)
##       alcohol              < 11.76667 to the left,  improve=0.06210660, (0 missing)
##       total.sulfur.dioxide < 67.5     to the left,  improve=0.04438619, (0 missing)
##       residual.sugar       < 1.375    to the left,  improve=0.02905351, (0 missing)
##       fixed.acidity        < 7.35     to the right, improve=0.02613259, (0 missing)
##   Surrogate splits:
##       total.sulfur.dioxide < 53.5     to the left,  agree=0.952, adj=0.214, (0 split)
##       volatile.acidity     < 0.875    to the right, agree=0.940, adj=0.024, (0 split)
##
```

```
## Node number 4: 1611 observations,    complexity param=0.01265926
##   mean=5.43203, MSE=0.5098121
##   left son=8 (688 obs) right son=9 (923 obs)
##   Primary splits:
##       volatile.acidity    < 0.3025   to the right, improve=0.04540111, (0 missing)
##       alcohol             < 10.05    to the left,  improve=0.03874403, (0 missing)
##       free.sulfur.dioxide < 13.5     to the left,  improve=0.03338886, (0 missing)
##       chlorides           < 0.0495   to the right, improve=0.02574623, (0 missing)
##       citric.acid         < 0.195    to the left,  improve=0.02327981, (0 missing)
##   Surrogate splits:
##       citric.acid          < 0.215    to the left,  agree=0.633, adj=0.141, (0 split)
##       free.sulfur.dioxide  < 20.5     to the left,  agree=0.600, adj=0.063, (0 split)
##       chlorides            < 0.0595   to the right, agree=0.593, adj=0.047, (0 split)
##       residual.sugar       < 1.15     to the left,  agree=0.583, adj=0.023, (0 split)
##       total.sulfur.dioxide < 219.25   to the right, agree=0.582, adj=0.022, (0 split)
##
## Node number 5: 761 observations
##   mean=5.971091, MSE=0.5878633
##
## Node number 6: 84 observations
##   mean=5.369048, MSE=1.137613
##
## Node number 7: 1294 observations,    complexity param=0.01970128
##   mean=6.391036, MSE=0.6894405
##   left son=14 (629 obs) right son=15 (665 obs)
##   Primary splits:
##       alcohol              < 11.76667 to the left,  improve=0.06504696, (0 missing)
##       chlorides            < 0.0395   to the right, improve=0.02758705, (0 missing)
##       fixed.acidity        < 7.35     to the right, improve=0.02750932, (0 missing)
##       pH                   < 3.055    to the left,  improve=0.02307356, (0 missing)
##       total.sulfur.dioxide < 191.5    to the right, improve=0.02186818, (0 missing)
##   Surrogate splits:
##       density              < 0.990885 to the right, agree=0.720, adj=0.424, (0 split)
##       volatile.acidity     < 0.2675   to the left,  agree=0.637, adj=0.253, (0 split)
##       chlorides            < 0.0365   to the right, agree=0.630, adj=0.238, (0 split)
##       residual.sugar       < 1.475    to the left,  agree=0.575, adj=0.126, (0 split)
##       total.sulfur.dioxide < 128.5    to the right, agree=0.574, adj=0.124, (0 split)
##
## Node number 8: 688 observations
##   mean=5.255814, MSE=0.4054895
##
## Node number 9: 923 observations
##   mean=5.56338, MSE=0.5471747
##
## Node number 14: 629 observations,    complexity param=0.01007193
##   mean=6.173291, MSE=0.6838017
##   left son=28 (11 obs) right son=29 (618 obs)
##   Primary splits:
##       volatile.acidity     < 0.465    to the right, improve=0.06897561, (0 missing)
##       total.sulfur.dioxide < 200      to the right, improve=0.04223066, (0 missing)
##       residual.sugar       < 0.975    to the left,  improve=0.03061714, (0 missing)
##       fixed.acidity        < 7.35     to the right, improve=0.02978501, (0 missing)
##       sulphates            < 0.575    to the left,  improve=0.02165970, (0 missing)
##   Surrogate splits:
```

```
##       citric.acid          < 0.045    to the left,  agree=0.986, adj=0.182, (0 split)
##       total.sulfur.dioxide < 279.25   to the right, agree=0.986, adj=0.182, (0 split)
##
## Node number 15: 665 observations
##   mean=6.596992, MSE=0.6075098
##
## Node number 28: 11 observations
##   mean=4.545455, MSE=0.9752066
##
## Node number 29: 618 observations
##   mean=6.202265, MSE=0.6306098
```

**Visualizing decision trees**

```r
rpart.plot(m.rpart, digits = 3)
```




## Step 4 – evaluating model performance

```r
p.rpart <- predict(m.rpart, wine_test)
```

A quick look at the summary statistics of our predictions suggests a potential problem: the predictions fall into a much narrower range than the true values:

```r
summary(p.rpart)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   4.545   5.563   5.971   5.893   6.202   6.597
```

```r
summary(wine_test$quality)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.000   5.000   6.000   5.901   6.000   9.000
```

This finding suggests that the model is not correctly identifying the extreme cases, and in particular, the best and worst wines.

The correlation between the predicted and actual quality values provides a simple way to gauge the model's performance.

```r
cor(p.rpart, wine_test$quality)
```

```
## [1] 0.5369525
```

A correlation of 0.54 is certainly acceptable. However, the correlation only measures how strongly the predictions are related to the true value; it is not a measure of how far off the predictions were from the true values.

**Measuring performance with the mean absolute error**

```r
MAE <- function(actual, predicted) {
  mean(abs(actual - predicted))
}

MAE(p.rpart, wine_test$quality)
```

```
## [1] 0.5872652
```

This implies that, on average, the difference between our model's predictions and the true quality score was about 0.59. On a quality scale from 0 to 10, this seems to suggest that our model is doing fairly well.

On the other hand, recall that most wines were neither very good nor very bad; the typical quality score was around 5 to 6. Therefore, a classifier that did nothing but predict the mean value may also do fairly well according to this metric.

```r
mean(wine_train$quality)
```

```
## [1] 5.870933
```

If we predicted the value 5.87 for every wine sample, we would have a MAE of only about 0.67:

```r
MAE(5.87, wine_test$quality)
```

```
## [1] 0.6722474
```

Our regression tree (MAE = 0.59) comes closer on average to the true quality score than the imputed mean (MAE = 0.67), but not by much. In comparison, Cortez reported an MAE of 0.58 for the neural network model and an MAE of 0.45 for the support vector machine. This suggests that there is room for improvement.

## Step 5 – improving model performance

To improve the performance of our learner, let's apply a model tree algorithm, which is a more complex application of trees to numeric prediction. Recall that a model tree extends regression trees by replacing the leaf nodes with regression models. This often results in more accurate results than regression trees, which use only a single numeric value for the prediction at the leaf nodes.

The current state-of-the-art in model trees is the **Cubist** algorithm, which itself is an enhance- ment of the M5 model tree algorithm

```r
library(Cubist)
```

```
## Lade nötiges Paket: lattice
```

```r
m.cubist <- cubist(x = wine_train[-12], y = wine_train$quality)
```

```r
m.cubist
```

```
##
## Call:
## cubist.default(x = wine_train[-12], y = wine_train$quality)
##
## Number of samples: 3750
## Number of predictors: 11
##
## Number of committees: 1
## Number of rules: 25
```

```r
summary(m.cubist)
```

```
##
## Call:
## cubist.default(x = wine_train[-12], y = wine_train$quality)
##
##
## Cubist [Release 2.07 GPL Edition]  Wed Feb 12 13:32:56 2025
## ---------------------------------
##
##     Target attribute 'outcome'
##
## Read 3750 cases (12 attributes) from undefined.data
##
## Model:
##
##   Rule 1: [21 cases, mean 5.0, range 4 to 6, est err 0.5]
##
##     if
##  free.sulfur.dioxide > 30
##  total.sulfur.dioxide > 195
##  total.sulfur.dioxide <= 235
##  sulphates > 0.64
##  alcohol > 9.1
```

```
##    then
##  outcome = 573.6 + 0.0478 total.sulfur.dioxide - 573 density
##           - 0.788 alcohol + 0.186 residual.sugar - 4.73 volatile.acidity
##
##   Rule 2: [28 cases, mean 5.0, range 4 to 8, est err 0.7]
##
##    if
##  volatile.acidity > 0.31
##  citric.acid <= 0.36
##  residual.sugar <= 1.45
##  total.sulfur.dioxide <= 97
##  alcohol > 9.1
##    then
##  outcome = 168.2 + 4.75 citric.acid + 0.0123 total.sulfur.dioxide
##           - 170 density + 0.057 residual.sugar - 6.4 chlorides + 0.84 pH
##           + 0.14 fixed.acidity
##
##   Rule 3: [171 cases, mean 5.1, range 3 to 6, est err 0.3]
##
##    if
##  volatile.acidity > 0.205
##  chlorides <= 0.054
##  density <= 0.99839
##  alcohol <= 9.1
##    then
##  outcome = 147.4 - 144 density + 0.08 residual.sugar + 0.117 alcohol
##           - 0.87 volatile.acidity - 0.09 pH - 0.01 fixed.acidity
##
##   Rule 4: [37 cases, mean 5.3, range 3 to 6, est err 0.5]
##
##    if
##  free.sulfur.dioxide > 30
##  total.sulfur.dioxide > 235
##  alcohol > 9.1
##    then
##  outcome = 19.5 - 0.013 total.sulfur.dioxide - 2.7 volatile.acidity
##           - 10 density + 0.005 residual.sugar + 0.008 alcohol
##
##   Rule 5: [64 cases, mean 5.3, range 5 to 6, est err 0.3]
##
##    if
##  volatile.acidity > 0.205
##  residual.sugar > 17.85
##    then
##  outcome = -23.6 + 0.233 alcohol - 5.2 chlorides - 0.75 citric.acid
##           + 28 density - 0.81 volatile.acidity - 0.19 pH
##           - 0.002 residual.sugar
##
##   Rule 6: [56 cases, mean 5.3, range 4 to 7, est err 0.6]
##
##    if
##  fixed.acidity <= 7.1
##  volatile.acidity > 0.205
##  chlorides > 0.054
```

```
##  density <= 0.99839
##  alcohol <= 9.1
##     then
##  outcome = 40.6 + 0.374 alcohol - 1.62 volatile.acidity
##            + 0.026 residual.sugar - 38 density - 0.21 pH
##            - 0.01 fixed.acidity
##
##   Rule 7: [337 cases, mean 5.3, range 3 to 7, est err 0.4]
##
##     if
##  fixed.acidity <= 7.8
##  volatile.acidity > 0.305
##  chlorides <= 0.09
##  free.sulfur.dioxide <= 82.5
##  total.sulfur.dioxide > 130
##  total.sulfur.dioxide <= 235
##  sulphates <= 0.64
##  alcohol <= 10.4
##     then
##  outcome = -32.1 + 0.233 alcohol - 9.7 chlorides
##            + 0.0038 total.sulfur.dioxide - 0.0081 free.sulfur.dioxide
##            + 35 density + 0.81 volatile.acidity
##
##   Rule 8: [30 cases, mean 5.5, range 3 to 7, est err 0.5]
##
##     if
##  fixed.acidity > 7.1
##  volatile.acidity > 0.205
##  chlorides > 0.054
##  density <= 0.99839
##  alcohol <= 9.1
##     then
##  outcome = 244 - 1.56 fixed.acidity - 228 density
##            + 0.0252 free.sulfur.dioxide - 7.3 chlorides
##            - 0.19 volatile.acidity + 0.003 residual.sugar
##
##   Rule 9: [98 cases, mean 5.5, range 4 to 8, est err 0.5]
##
##     if
##  volatile.acidity > 0.155
##  chlorides > 0.09
##  total.sulfur.dioxide <= 235
##  sulphates <= 0.64
##     then
##  outcome = 55.9 - 3.85 volatile.acidity - 52 density
##            + 0.023 residual.sugar + 0.092 alcohol + 0.35 pH
##            + 0.05 fixed.acidity + 0.3 sulphates
##            + 0.001 free.sulfur.dioxide
##
##   Rule 10: [446 cases, mean 5.6, range 4 to 8, est err 0.5]
##
##     if
##  fixed.acidity <= 7.8
##  volatile.acidity > 0.155
```

```
##   volatile.acidity <= 0.305
##   chlorides <= 0.09
##   free.sulfur.dioxide <= 82.5
##   total.sulfur.dioxide > 130
##   total.sulfur.dioxide <= 235
##   sulphates <= 0.64
##   alcohol > 9.1
##   alcohol <= 10.4
##      then
##   outcome = 15.1 + 0.35 alcohol - 3.09 volatile.acidity - 14.7 chlorides
##             + 1.16 sulphates - 0.0022 total.sulfur.dioxide
##             + 0.11 fixed.acidity + 0.45 pH + 0.5 citric.acid - 14 density
##             + 0.006 residual.sugar
##
##    Rule 11: [31 cases, mean 5.6, range 3 to 8, est err 0.8]
##
##      if
##   volatile.acidity > 0.31
##   citric.acid > 0.36
##   free.sulfur.dioxide <= 30
##   total.sulfur.dioxide <= 97
##      then
##   outcome = 3.2 + 0.0584 total.sulfur.dioxide + 7.77 volatile.acidity
##             + 0.328 alcohol - 9 density + 0.003 residual.sugar
##
##    Rule 12: [20 cases, mean 5.7, range 3 to 8, est err 0.9]
##
##      if
##   free.sulfur.dioxide > 82.5
##   total.sulfur.dioxide <= 235
##   sulphates <= 0.64
##   alcohol > 9.1
##      then
##   outcome = -8.9 + 109.3 chlorides + 0.948 alcohol
##
##    Rule 13: [331 cases, mean 5.8, range 4 to 8, est err 0.5]
##
##      if
##   volatile.acidity > 0.31
##   free.sulfur.dioxide <= 30
##   total.sulfur.dioxide > 97
##   alcohol > 9.1
##      then
##   outcome = 89.8 + 0.0234 free.sulfur.dioxide + 0.324 alcohol
##             + 0.07 residual.sugar - 90 density - 1.47 volatile.acidity
##             + 0.48 pH
##
##    Rule 14: [116 cases, mean 5.8, range 3 to 8, est err 0.6]
##
##      if
##   fixed.acidity > 7.8
##   volatile.acidity > 0.155
##   free.sulfur.dioxide > 30
##   total.sulfur.dioxide > 130
```

```
##  total.sulfur.dioxide <= 235
##  sulphates <= 0.64
##  alcohol > 9.1
##     then
##  outcome = 6 + 0.346 alcohol - 0.41 fixed.acidity - 1.69 volatile.acidity
##            - 2.9 chlorides + 0.19 sulphates + 0.07 pH
##
##   Rule 15: [115 cases, mean 5.8, range 4 to 7, est err 0.5]
##
##     if
##  volatile.acidity > 0.205
##  residual.sugar <= 17.85
##  density > 0.99839
##  alcohol <= 9.1
##     then
##  outcome = -110.2 + 120 density - 3.46 volatile.acidity - 0.97 pH
##            - 0.022 residual.sugar + 0.088 alcohol - 0.6 citric.acid
##            - 0.01 fixed.acidity
##
##   Rule 16: [986 cases, mean 5.9, range 3 to 9, est err 0.6]
##
##     if
##  volatile.acidity <= 0.31
##  free.sulfur.dioxide <= 30
##  alcohol > 9.1
##     then
##  outcome = 280.4 - 282 density + 0.128 residual.sugar
##            + 0.0264 free.sulfur.dioxide - 3 volatile.acidity + 1.2 pH
##            + 0.65 citric.acid + 0.09 fixed.acidity + 0.56 sulphates
##            + 0.015 alcohol
##
##   Rule 17: [49 cases, mean 6.0, range 5 to 8, est err 0.5]
##
##     if
##  volatile.acidity > 0.155
##  residual.sugar > 8.8
##  free.sulfur.dioxide > 30
##  total.sulfur.dioxide <= 130
##  pH <= 3.26
##  alcohol > 9.1
##     then
##  outcome = 173.5 - 169 density + 0.055 alcohol + 0.38 sulphates
##            + 0.002 residual.sugar
##
##   Rule 18: [114 cases, mean 6.1, range 3 to 9, est err 0.6]
##
##     if
##  volatile.acidity > 0.31
##  citric.acid <= 0.36
##  residual.sugar > 1.45
##  total.sulfur.dioxide <= 97
##  alcohol > 9.1
##     then
##  outcome = 302.3 - 305 density + 0.0128 total.sulfur.dioxide
```

```
##           + 0.096 residual.sugar + 1.94 citric.acid + 1.05 pH
##           + 0.17 fixed.acidity - 6.7 chlorides
##           + 0.0022 free.sulfur.dioxide - 0.21 volatile.acidity
##           + 0.013 alcohol + 0.09 sulphates
##
##   Rule 19: [145 cases, mean 6.1, range 5 to 8, est err 0.6]
##
##     if
##  volatile.acidity > 0.155
##  free.sulfur.dioxide > 30
##  total.sulfur.dioxide <= 195
##  sulphates > 0.64
##     then
##  outcome = 206 - 209 density + 0.069 residual.sugar + 0.38 fixed.acidity
##           + 2.79 sulphates + 0.0155 free.sulfur.dioxide
##           - 0.0051 total.sulfur.dioxide - 1.71 citric.acid + 1.04 pH
##
##   Rule 20: [555 cases, mean 6.1, range 3 to 9, est err 0.6]
##
##     if
##  total.sulfur.dioxide > 130
##  total.sulfur.dioxide <= 235
##  sulphates <= 0.64
##  alcohol > 10.4
##     then
##  outcome = 108 + 0.276 alcohol - 109 density + 0.05 residual.sugar
##           + 0.77 pH - 1.02 volatile.acidity - 4.2 chlorides
##           + 0.78 sulphates + 0.08 fixed.acidity
##           + 0.0016 free.sulfur.dioxide - 0.0003 total.sulfur.dioxide
##
##   Rule 21: [73 cases, mean 6.2, range 4 to 8, est err 0.4]
##
##     if
##  volatile.acidity > 0.155
##  citric.acid <= 0.28
##  residual.sugar <= 8.8
##  free.sulfur.dioxide > 30
##  total.sulfur.dioxide <= 130
##  pH <= 3.26
##  sulphates <= 0.64
##  alcohol > 9.1
##     then
##  outcome = 4.2 + 0.147 residual.sugar + 0.47 alcohol + 3.75 sulphates
##           - 2.5 volatile.acidity - 5 density
##
##   Rule 22: [244 cases, mean 6.3, range 4 to 8, est err 0.6]
##
##     if
##  citric.acid > 0.28
##  residual.sugar <= 8.8
##  free.sulfur.dioxide > 30
##  total.sulfur.dioxide <= 130
##  pH <= 3.26
##     then
```

```
##  outcome = 40.1 + 0.278 alcohol + 1.3 sulphates - 39 density
##            + 0.017 residual.sugar + 0.001 total.sulfur.dioxide + 0.17 pH
##            + 0.03 fixed.acidity
##
##   Rule 23: [106 cases, mean 6.3, range 4 to 8, est err 0.6]
##
##     if
##  volatile.acidity <= 0.155
##  free.sulfur.dioxide > 30
##     then
##  outcome = 139.1 - 138 density + 0.058 residual.sugar + 0.71 pH
##            + 0.92 sulphates + 0.11 fixed.acidity - 0.73 volatile.acidity
##            + 0.055 alcohol - 0.0012 total.sulfur.dioxide
##            + 0.0007 free.sulfur.dioxide
##
##   Rule 24: [137 cases, mean 6.5, range 4 to 9, est err 0.6]
##
##     if
##  volatile.acidity > 0.155
##  free.sulfur.dioxide > 30
##  total.sulfur.dioxide <= 130
##  pH > 3.26
##  sulphates <= 0.64
##  alcohol > 9.1
##     then
##  outcome = 114.2 + 0.0142 total.sulfur.dioxide - 107 density
##            - 11.8 chlorides - 1.57 pH + 0.124 alcohol + 1.21 sulphates
##            + 1.16 volatile.acidity + 0.021 residual.sugar
##            + 0.04 fixed.acidity
##
##   Rule 25: [92 cases, mean 6.5, range 4 to 8, est err 0.6]
##
##     if
##  volatile.acidity <= 0.205
##  alcohol <= 9.1
##     then
##  outcome = -200.7 + 210 density + 5.88 volatile.acidity + 23.9 chlorides
##            - 2.83 citric.acid - 1.17 pH
##
##
## Evaluation on training data (3750 cases):
##
##     Average  |error|                0.5
##     Relative |error|               0.67
##     Correlation coefficient        0.66
##
##
##  Attribute usage:
##    Conds  Model
##
##     84%    93%    alcohol
##     80%    89%    volatile.acidity
##     70%    61%    free.sulfur.dioxide
##     63%    50%    total.sulfur.dioxide
```

```
##     44%    70%    sulphates
##     26%    44%    chlorides
##     22%    76%    fixed.acidity
##     16%    87%    residual.sugar
##     11%    86%    pH
##     11%    45%    citric.acid
##      8%    97%    density
##
##
## Time: 0.4 secs
```

output:

" Rule 1: [21 cases, mean 5.0, range 4 to 6, est err 0.5]

```
if
free.sulfur.dioxide > 30
total.sulfur.dioxide > 195
total.sulfur.dioxide <= 235
sulphates > 0.64
alcohol > 9.1
then
outcome = **573.6 + 0.0478 total.sulfur.dioxide - 573 density**
          **- 0.788 alcohol + 0.186 residual.sugar - 4.73 volatile.acidity**
```

...

Rule 25: [92 cases, mean 6.5, range 4 to 8, est err 0.6]

```
if
volatile.acidity <= 0.205
alcohol <= 9.1
then
outcome = **-200.7 + 210 density + 5.88 volatile.acidity + 23.9 chlorides**
          **- 2.83 citric.acid - 1.17 pH**
```

Evaluation on training data (3750 cases):

```
Average  |error|                0.5
Relative |error|               0.67
Correlation coefficient        0.66


Attribute usage:
  Conds  Model

   84%    93%    alcohol
   80%    89%    volatile.acidity
   70%    61%    free.sulfur.dioxide
   63%    50%    total.sulfur.dioxide
   44%    70%    sulphates
```

```
26%     44%     chlorides
22%     76%     fixed.acidity
16%     87%     residual.sugar
11%     86%     pH
11%     45%     citric.acid
 8%     97%     density
```

"

the if portion of the output is somewhat like the regression tree we built earlier. A series of decisions based on the wine properties of sulfur dioxide, sulphates, and alcohol creates a rule culminating in the final prediction. A key difference between this model tree output and the earlier regression tree output, however, is that **the nodes here terminate not in a numeric prediction, but rather in a linear model.**

It is important to note that the regression effects estimated by this model apply only to wine samples reaching this node

To examine the performance of this model, we'll look at how well it performs on the unseen test data. The predict() function gets us a vector of predicted values:

```
p.cubist <- predict(m.cubist, wine_test)

summary(p.cubist)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.677   5.416   5.906   5.848   6.238   7.393
```

The model tree appears to be predicting a wider range of values than the regression tree

Correlation:

```
cor(p.cubist, wine_test$quality)
```

```
## [1] 0.6201015
```

Furthermore, the model slightly reduced the MAE:

```
MAE(wine_test$quality, p.cubist)
```

```
## [1] 0.5339725
```

Although we did not improve a great deal beyond the regression tree, we surpassed the perfor- mance of the neural network model published by Cortez, and we are getting closer to the pub- lished MAE value of 0.45 for the support vector machine model, all while using a much simpler learning method.