ADA 442 HomeWork

Homework 4: Tree Based Models

Furkan

1/6/2022

Contents

1 Introduction

- The ultimate purpose of this research is to observe how Pruned and Unpruned Tree's behave on 'OJ' Data set.
- Also, Finding optimal tree size and its impact can be other achievement.

2 Methodology

- To predict data, 'tree' library has used.
- To find error rate, (TP + TN)/Size formula has used mentally. -even when algorithm has been used-
- To observe relation between variables or values, confusion matrices and plots have been used.

3 Data Set

```
library(ISLR2)
set.seed(73745) # for reproducible results
data("OJ")
```

- I have used Orange Juice Data which contains 1070 purchases where the customer either purchased Citrus Hill or Minute Maid Orange Juice from 'ISLR2' library.
- OJ data frame has consisted of 1070 observations on the following 18 variables.

4 Explaratory Data analysis

Brief information about the data set can be seen below

```
head(OJ)
```

```
##   Purchase WeekofPurchase StoreID PriceCH PriceMM DiscCH DiscMM SpecialCH
## 1       CH            237       1    1.75    1.99   0.00    0.0         0
## 2       CH            239       1    1.75    1.99   0.00    0.3         0
## 3       CH            245       1    1.86    2.09   0.17    0.0         0
## 4       MM            227       1    1.69    1.69   0.00    0.0         0
## 5       CH            228       7    1.69    1.69   0.00    0.0         0
## 6       CH            230       7    1.69    1.99   0.00    0.0         0
##   SpecialMM  LoyalCH SalePriceMM SalePriceCH PriceDiff Store7 PctDiscMM
## 1         0 0.500000        1.99        1.75      0.24     No  0.000000
## 2         1 0.600000        1.69        1.75     -0.06     No  0.150754
## 3         0 0.680000        2.09        1.69      0.40     No  0.000000
## 4         0 0.400000        1.69        1.69      0.00     No  0.000000
## 5         0 0.956535        1.69        1.69      0.00    Yes  0.000000
## 6         1 0.965228        1.99        1.69      0.30    Yes  0.000000
##   PctDiscCH ListPriceDiff STORE
## 1  0.000000          0.24     1
## 2  0.000000          0.24     1
## 3  0.091398          0.23     1
## 4  0.000000          0.00     1
## 5  0.000000          0.00     0
## 6  0.000000          0.30     0
```

```
summary(OJ)
```

```
##  Purchase WeekofPurchase     StoreID        PriceCH         PriceMM     
##  CH:653   Min.   :227.0   Min.   :1.00   Min.   :1.690   Min.   :1.690  
##  MM:417   1st Qu.:240.0   1st Qu.:2.00   1st Qu.:1.790   1st Qu.:1.990  
##           Median :257.0   Median :3.00   Median :1.860   Median :2.090  
##           Mean   :254.4   Mean   :3.96   Mean   :1.867   Mean   :2.085  
##           3rd Qu.:268.0   3rd Qu.:7.00   3rd Qu.:1.990   3rd Qu.:2.180  
##           Max.   :278.0   Max.   :7.00   Max.   :2.090   Max.   :2.290  
##      DiscCH            DiscMM         SpecialCH        SpecialMM     
##  Min.   :0.00000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
##  1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000  
##  Median :0.00000   Median :0.0000   Median :0.0000   Median :0.0000  
##  Mean   :0.05186   Mean   :0.1234   Mean   :0.1477   Mean   :0.1617  
##  3rd Qu.:0.00000   3rd Qu.:0.2300   3rd Qu.:0.0000   3rd Qu.:0.0000  
##  Max.   :0.50000   Max.   :0.8000   Max.   :1.0000   Max.   :1.0000  
##     LoyalCH          SalePriceMM     SalePriceCH      PriceDiff       Store7 
##  Min.   :0.000011   Min.   :1.190   Min.   :1.390   Min.   :-0.6700   No :714  
##  1st Qu.:0.325257   1st Qu.:1.690   1st Qu.:1.750   1st Qu.: 0.0000   Yes:356  
##  Median :0.600000   Median :2.090   Median :1.860   Median : 0.2300            
##  Mean   :0.565782   Mean   :1.962   Mean   :1.816   Mean   : 0.1465            
##  3rd Qu.:0.850873   3rd Qu.:2.130   3rd Qu.:1.890   3rd Qu.: 0.3200            
##  Max.   :0.999947   Max.   :2.290   Max.   :2.090   Max.   : 0.6400            
##    PctDiscMM        PctDiscCH       ListPriceDiff       STORE      
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.000   Min.   :0.000  
##  1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.140   1st Qu.:0.000  
##  Median :0.0000   Median :0.00000   Median :0.240   Median :2.000  
##  Mean   :0.0593   Mean   :0.02731   Mean   :0.218   Mean   :1.631  
##  3rd Qu.:0.1127   3rd Qu.:0.00000   3rd Qu.:0.300   3rd Qu.:3.000  
##  Max.   :0.4020   Max.   :0.25269   Max.   :0.440   Max.   :4.000  
```

5 Model Fit

- Data set has been divided into two part as 80% and 20%.

```
index = sample(1:nrow(OJ), 0.8*nrow(OJ))  # %80 for training, %20 for testing
train = OJ[index,] # Create the training data
test = OJ[-index,] # Create the test data
dim(train)
```

```
## [1] 856  18
```

```
dim(test)
```

```
## [1] 214  18
```

5.1 *Fit the Tree to the training data and Obtain Summary Statistics*

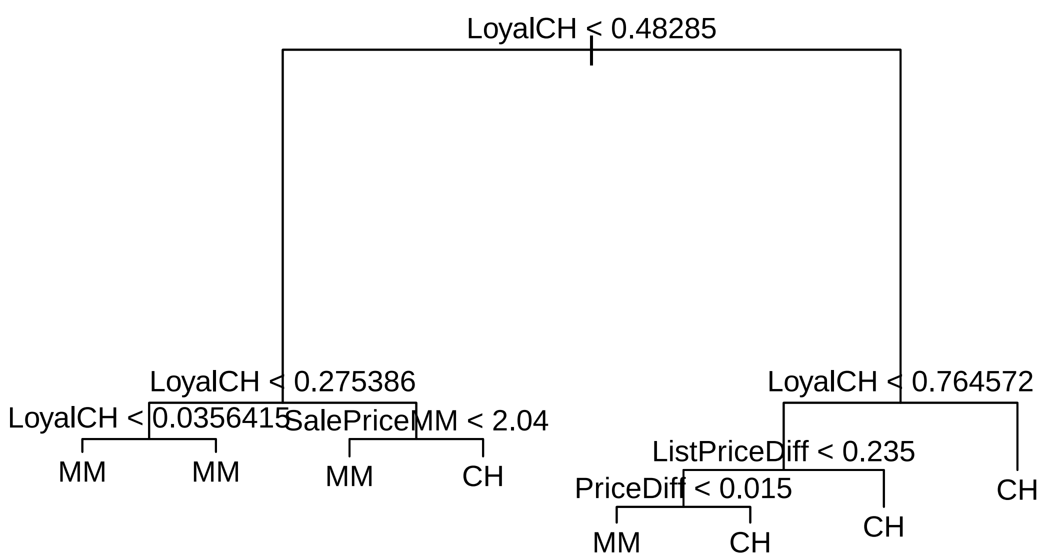
```
library(tree)
tree_model <- tree(Purchase ~ ., train)
summary(tree_model)
```

```
##
## Classification tree:
## tree(formula = Purchase ~ ., data = train)
## Variables actually used in tree construction:
## [1] "LoyalCH"       "SalePriceMM"   "ListPriceDiff" "PriceDiff"
## Number of terminal nodes:  8
## Residual mean deviance:  0.7589 = 643.6 / 848
## Misclassification error rate: 0.1554 = 133 / 856
```

- The number of terminal nodes is 9.
- The Missclassification error rate is 16.12%
- There are 5 variables that used in tree construction which are "LoyalCH", "SalePriceMM", "SpecialCH", "ListPriceDiff" and "SalePriceCH".

5.2 *Demonstration of the Tree and interpretation of results*

```
plot(tree_model)
text(tree_model, pretty = 0, cex = 0.9)
```



- We can say that "LoyalCH" is the most significant variable because first and second decisions will be made by using that variable.

5.3 *Confusion Matrix of the Tree*

```r
test_pred <- predict(tree_model, test, type = "class")
table(test_pred, test_actual = test$Purchase)
```

```
##          test_actual
## test_pred  CH  MM
##        CH 111  21
##        MM  15  67
```

- As seen confusion matrix above, we see that 114 of Citrus Hill were correctly classified and 53 of Minute Maid were correctly classified.
- *So, our prediction accuracy is:*

```r
pred_accuracy <- (114+53)/214
pred_accuracy
```

```
## [1] 0.7803738
```

```r
# So the error rate is simply (1 - pred_accuracy)
1 - mean(test_pred == test$Purchase)
```

```
## [1] 0.1682243
```

5.4 *Optimal Tree Size*

- If we use 5 folds and 10 different values of alpha we'll build 50 different trees. For each value of alpha we'll split the data into 5 folds and build 5 trees, using 4 folds to train and the left out fold to get a mean squared prediction error.
- The ultimate aim is to find the value with the lowest average MSE.

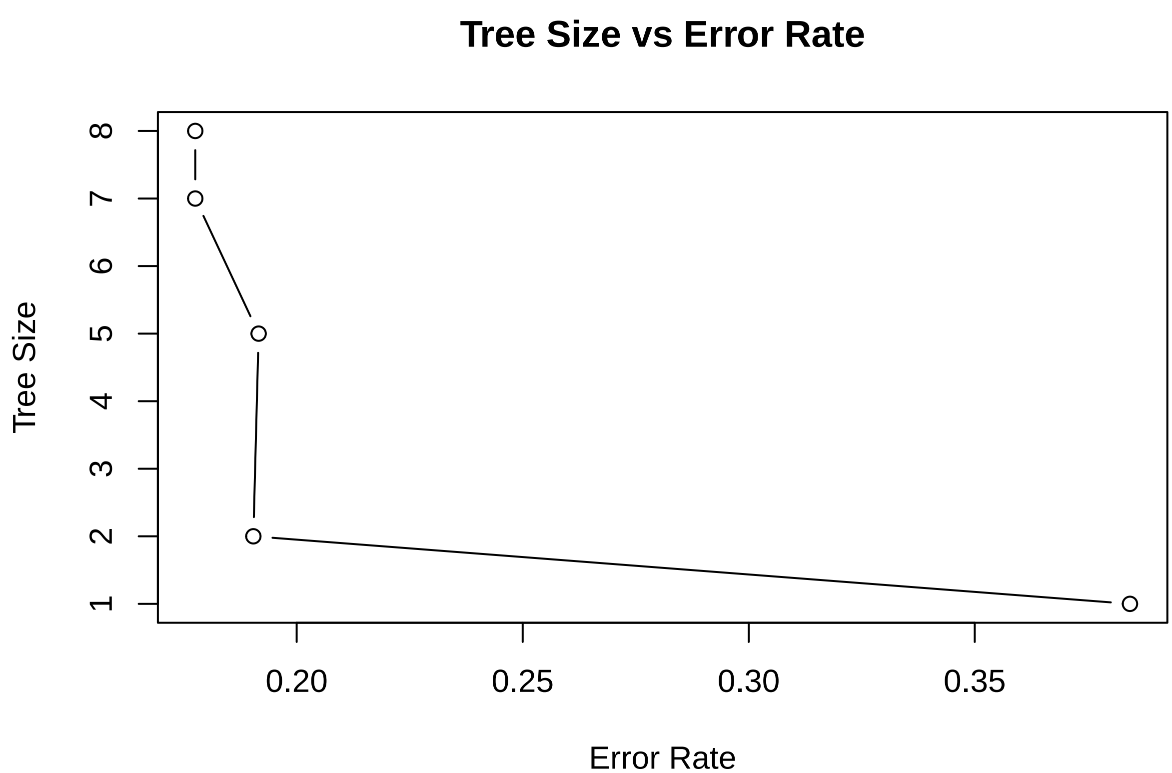
```r
cv_tree_model <- cv.tree(tree_model, FUN = prune.misclass)
cv_tree_model
```

```
## $size
## [1] 8 7 5 2 1
##
## $dev
## [1] 152 152 164 163 329
##
## $k
## [1]  -Inf   0.0   4.5   6.0 169.0
##
## $method
## [1] "misclass"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

- k = nagative Inf is allowing a full, unpruned tree. Where k=159 which is the the highest value in our results corresponds to a single node tree.
- We make our selection based on "dev" because we've changed the pruning function, this is actually the number of misclassified values. Lower is better and the minimum dev value is 151. That corresponds to a tree with 9 or 5 terminal nodes and alpha of -Inf or 0.0.

5.5 Producing a plot with tree size on the x-axis and cross-validated classification error rate on the y-axis

```
CV_Error <- (cv_tree_model$dev / nrow(train))
plot(CV_Error, cv_tree_model$size, type = "b",
      xlab = "Error Rate",
     ylab = "Tree Size",
     main = "Tree Size vs Error Rate")
```



- Tree has already has 9 terminal nodes so 5 terminal nodes will be used at following steps.

- However, 9 or 5 can be chosen as optimal tree size.

5.6 *Producing Pruned Tree with optimal tree size*

```r
pruned_tree_model <- prune.tree(tree_model, best = 5)
summary(pruned_tree_model)
```

```
##
## Classification tree:
## snip.tree(tree = tree_model, nodes = c(4L, 12L, 5L))
## Variables actually used in tree construction:
## [1] "LoyalCH"       "ListPriceDiff"
## Number of terminal nodes:  5
## Residual mean deviance:  0.8059 = 685.8 / 851
## Misclassification error rate: 0.1869 = 160 / 856
```

```r
test_pred_prunned <- predict(pruned_tree_model, test, type = "class")
table(test_pred_prunned, test_actual = test$Purchase)
```

```
##                  test_actual
## test_pred_prunned CH MM
##                CH 99 19
##                MM 27 69
```

- *So, our pruned prediction accuracy is:*

```r
pred_accuracy_prunned <- mean(test_pred_prunned == test$Purchase)
pred_accuracy_prunned
```

```
## [1] 0.7850467
```

```r
# So the error rate is simply (1 - pred_accuracy)
1 - mean(test_pred_prunned == test$Purchase)
```

```
## [1] 0.2149533
```

6 Overall Comparison

- *Training Error Rates between Pruned and Unpruned Trees:*

```r
# Unpruned Tree :
mean(predict(tree_model, type = "class") != train$Purchase)
```

```
## [1] 0.1553738
```

```r
# Pruned Tree :
mean(predict(pruned_tree_model, type = "class") != train$Purchase)
```

```
## [1] 0.1845794
```

- *Test Error Rates between Pruned and Unpruned Trees:*

```r
# Unpruned Tree :
mean(predict(tree_model, type = "class", newdata = test) != test$Purchase)
```

```
## [1] 0.1682243
```

```r
# Pruned Tree :
mean(predict(pruned_tree_model, type = "class", newdata = test) != test$Purchase)
```

```
## [1] 0.1728972
```

7 Conclusion

- *Error Rate of Unpruned Tree :* 0.2196262
- *Error Rate of Pruned Tree :* 0.2149533
- There is an improvement in respect to Error Rates. However, As stated above, there were no difference between dev values for 9 terminal nodes and 5 terminal nodes.

-Improvement been considered improvement but using Unpruned Tree is enough for 'OJ' data set. Also, possibly there are better algorithms for prediction. So, other possibilities can be considered if data set is suitable.

8 References

- Lecture Slides
- https://rstudio-pubs-static.s3.amazonaws.com/442284_82321e66af4e49d58adcd897e00bf495.html
- https://chirag-sehra.medium.com/decision-trees-explained-easily-28f23241248
- https://rpubs.com/miss_kris/795888