Home Work Data Analysis 1

Md Shamsuzzaman

11.15.2017

1 Executive Summary

The OJ data is consisted of 1070 purchases which has been recorded on the basis of purchasing either Citrus hill or Minute maid orange juice along with a number of characteristics of customers and products. The tree based method and for regression and classification has been used for the analysis of the data set. Through this analysis, segmenting of the predictor space has been turned into a number of simple regions along with the identification of the prediction for the class or region of the newcomers. Basically, it provides decision on the prediction about an additional new observation. At the beginning in this study on the OJ data set, a training data set which is consisted of a random sample of 800 observations was created along with the a test set containing the remaining observations. Then, the training data has been fitted to a tree with purchase as the response and the other variables except Buy as predictors. A training error rate or misclassification rate has been carried out from the fitted tree. After that, the predictions of the responses along with confusion matrix have been made through the fitted tree. This attained confusion matrix has reveals the misclassification of the data set. However, a pruning of tree has been performed by using cross validation to get the best terminal nodes to create a simpler model of prediction and thus improving the performance. Finally, a comparison of the error rates has been identified between the pruned and the unpruned tress. The better tree model was concluded according to the low value of the misclassification rate. The analysis, of the training data, has indicated 7 terminal nodes for the tree and the error rate was 15.75% with 20% misclassification rate. The cross-validated classification applications revealed that the lowest deviance (701.2572) was on 5 and 6 terminal nodes. However, 5 terminal nodes has been used to prune the tree due to the more simplicity of 5 nodes rather than 6. Moreover, the training error rates of the pruned and unpruned trees were 15.875% and 15.75% respectively with less variability between them. On the other hand, the test error rates of the pruned and unpruned tress were 19.62% and 20% respectively. So, the test error rate of the pruned data set has a lower (by 0.4%) misclassification rate then the unpruned tree.

2 Introduction

The OJ data is consisted of 1070 purchases which has been recorded on the basis of purchasing either Citrus hill or Minute maid orange juice along with a number of characteristics of customers and products. The objective of this study was to create a model prediction for the decision of the customer to buy either Citrus Hill or Minute Maid based on a number of characteristics of the customer and product. The tree based method and for regression and classification has been used for the analysis of the data set.

3 Data Collection

The OJ data is consisted of 1070 observations along with 18 variables. The variables are purchase - a factor with levels CH and MM indicating whether the customer purchased Citrus Hill or Minute Maid Orange Juice, week of Purchase - week of purchase, store ID ? ID of the Store, priceCH - price charged for CH, priceMM - price charged for MM, discCH - discount offered for CH, discMM - discount offered for MM, specialCH - indicator of special on CH, specialMM - indicator of special on MM, loyalCH - customer brand loyalty for CH, salePriceMM - sale price for MM, salePriceCH - sale price for CH, priceDiff - sale price of MM less sale price of CH, store7 - a factor with levels No and Yes indicating whether the sale is at Store 7, pctDiscMM : Percentage discount for MM, pctDiscCH - percentage discount for CH, listPriceDiff - List price of MM less list price of CH, and STORE - which of 5 possible stores the sale occurred.

4 Summary Details

Different visual presentations have been used for the OJ data set

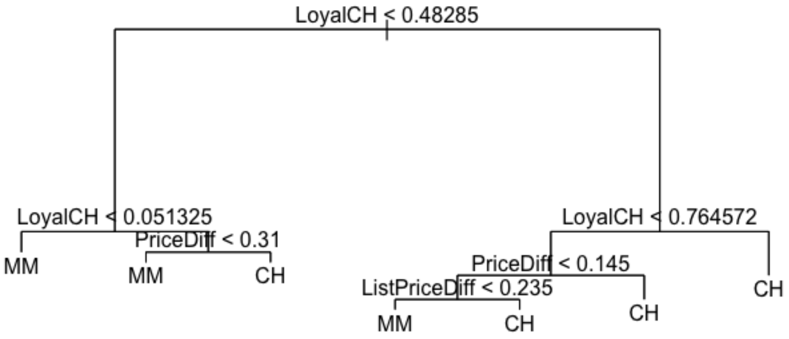

Fig 1. Regression Tree

7 terminal nodes are revealed for 3 variables (loyalCH, pricediff, and listpricediff) through the tree based analysis on training data set. If the loyalCH, at the last terminal of the tree, is greater than 0.764572, the customer is loyal to Citrus hill orange juice with the trend to buy.

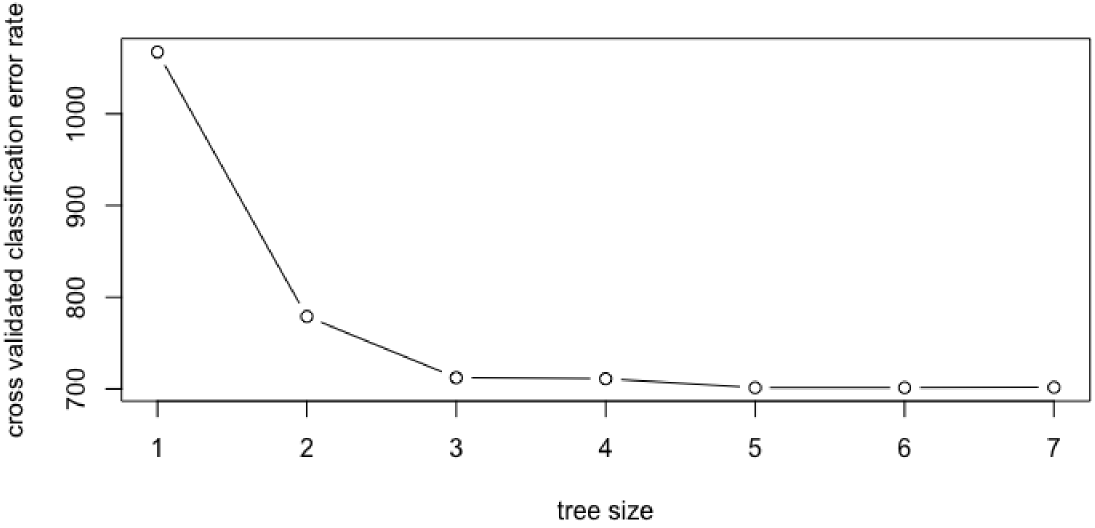

Fig 2. Cross-validated plot for training data set

From the cross-validated plot, the lowest cross-validated classification error rates are on the terminals of 5 and 6. Due to the more simplicity, 5 terminal node has been used for pruning the tree.

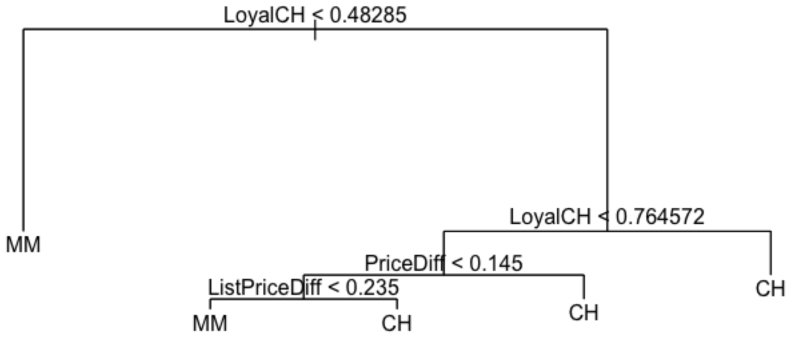

Fig 3. Pruned tree for OJ data set

The pruned tree has shown the simpler model with 5 terminal node. According to the pruned tree, if the loyalCH is smaller than 0.48285, the customer does not want to purchase the Citrus hill orange juice.

5 Analysis

```
> library(ISLR)
> attach(OJ)
> library(tree)
> library(MASS)
> set.seed(3)
> tr<-sample(1:nrow(OJ),800)
> oj.tr<-OJ[tr,]
> oj.te<-OJ[-tr,]
> tree.oj<-tree(Purchase~.,oj.tr)
> summary(tree.oj)

Classification tree:
tree(formula = Purchase ~ ., data = oj.tr)
Variables actually used in tree construction:
[1] "LoyalCH"         "PriceDiff"     "ListPriceDiff"
Number of terminal nodes:  7
Residual mean deviance:  0.7425 = 588.8 / 793
Misclassification error rate: 0.1575 = 126 / 800

> tree.oj
node), split, n, deviance, yval, (yprob)
      * denotes terminal node

 1) root 800 1064.00 CH ( 0.61750 0.38250 )
   2) LoyalCH < 0.48285 290  308.60 MM ( 0.22414 0.77586 )
     4) LoyalCH < 0.051325 55    0.00 MM ( 0.00000 1.00000 ) *
     5) LoyalCH > 0.051325 235  277.20 MM ( 0.27660 0.72340 )
      10) PriceDiff < 0.31 186  193.70 MM ( 0.21505 0.78495 ) *
      11) PriceDiff > 0.31 49   67.91 CH ( 0.51020 0.48980 ) *
   3) LoyalCH > 0.48285 510  446.50 CH ( 0.84118 0.15882 )
     6) LoyalCH < 0.764572 244  294.30 CH ( 0.70902 0.29098 )
      12) PriceDiff < 0.145 96  133.00 MM ( 0.48958 0.51042 )
        24) ListPriceDiff < 0.235 65   84.47 MM ( 0.35385 0.64615 ) *
        25) ListPriceDiff > 0.235 31   33.12 CH ( 0.77419 0.22581 ) *
      13) PriceDiff > 0.145 148  124.40 CH ( 0.85135 0.14865 ) *
     7) LoyalCH > 0.764572 266   85.24 CH ( 0.96241 0.03759 ) *
```

7 nodes through 3 variables, LoyalCH, PriceDiff, and ListPriceDiff, have been found from the summary of the fitted tree of the training data set with 15.75% error rates.

```
> full.te.predict<-predict(tree.oj,oj.te, type="class")
> table(full.te.predict, oj.te$Purchase)

full.te.predict  CH  MM
             CH 132  27
             MM  27  84
> mean(full.te.predict!=oj.te$Purchase)
[1] 0.2
```

132 observations of test data set are correctly predicted to decide to buy Citrus Hill, and 84 observations are correctly predicted or classified to buy Minute Maid from the confusion matrix with a 20% misclassification. Where the MSE value of the test data set is 0.2. Now, the cv.tree function has been applied to determine the optimal tree size or the best nodes for a simpler model.

```
> cv.oj1=cv.tree(tree.oj)
> cv.oj1
$size
[1] 7 6 5 4 3 2 1

$dev
[1]  701.6317  701.2572  701.2572  711.0494  712.2034  778.9619 1067.5296

$k
[1]      -Inf  15.45135  15.60109  31.45203  36.81015  66.95200 309.35604

$method
[1] "deviance"

attr(,"class")
[1] "prune"         "tree.sequence"
```

It has been found from the cv.tree function, the deviance values are lowest at the terminal nodes of 5 and 6. Terminal node of 5 has been used for the pruning tree due to the more simplicity from 6. Now the pruning of the tree has been done for the pruned and unpruned tree to calculate the training and test error rates.

```
> prune.oj<-prune.misclass(tree.oj,best=5)
> plot(prune.oj)
> text(prune.oj, pretty=0)
> mean(predict(prune.oj,oj.tr,type="class")!=oj.tr$Purchase)
[1] 0.15875
> mean(predict(prune.oj, oj.te,type="class")!=oj.te$Purchase)
[1] 0.1962963
```

From the analysis, the training error rates of the pruned and unpruned trees are 15.875% and 15.75% respectively with less variability between them. On the other hand, the test error rates of the pruned and unpruned tress are 19.62% and 20% respectively. So, the test error rate of the pruned data set has a lower (by 0.4%) misclassification rate then the unpruned tree.

6 Conclusion

The better tree model has been concluded according to the low value of the misclassification rate. The analysis, of the training data, has indicated 7 terminal nodes for the tree and the error rate was 15.75% with 20% misclassification rate. The cross-validated classification applications revealed that the lowest deviance (701.2572) was on 5 and 6 terminal nodes. However, 5 terminal nodes has been used to prune the tree due to the more simplicity of 5 nodes rather than 6. It can be concluded that the pruning tree by using 5 terminal nodes increases the performance of the prediction model of the test data set.

7 Appendix

7.1 R Code

```
library(ISLR)
attach(OJ)
library(tree)
library(MASS)
set.seed(3)
tr<-sample(1:nrow(OJ),800)
oj.tr<-OJ[tr,]
oj.te<-OJ[-tr,]
tree.oj<-tree(Purchase~.,oj.tr)
summary(tree.oj)
tree.oj
plot(tree.oj, main="OJ Purchase Decision Tree")
text(tree.oj, pretty=0)
```

```
full.te.predict<-predict(tree.oj,oj.te, type="class")
table(full.te.predict, oj.te$Purchase)
mean(full.te.predict!=oj.te$Purchase)
cv.oj1=cv.tree(tree.oj)
cv.oj1
plot(cv.oj1$size,cv.oj1$dev,type="b", xlab="tree size",ylab="cross validated classification
prune.oj<-prune.misclass(tree.oj,best=5)
plot(prune.oj)
text(prune.oj, pretty=0)
mean(predict(prune.oj,oj.tr,type="class")!=oj.tr$Purchase)
mean(predict(prune.oj, oj.te,type="class")!=oj.te$Purchase)
```

7.2 Log File

```
> library("ISLR", lib.loc="/Library/Frameworks/R.framework/Versions/3.3/Resources/library")
> library(ISLR)
> attach(OJ)
> library(tree)
> library(MASS)
> set.seed(3)
> tr<-sample(1:nrow(OJ),800)
> oj.tr<-OJ[tr,]
> oj.te<-OJ[-tr,]
> tree.oj<-tree(Purchase˜.,oj.tr)
> summary(tree.oj)

Classification tree:
tree(formula = Purchase ˜ ., data = oj.tr)
Variables actually used in tree construction:
[1] "LoyalCH"       "PriceDiff"     "ListPriceDiff"
Number of terminal nodes:  7
Residual mean deviance:  0.7425 = 588.8 / 793
Misclassification error rate: 0.1575 = 126 / 800
> tree.oj
node), split, n, deviance, yval, (yprob)
      * denotes terminal node

 1) root 800 1064.00 CH ( 0.61750 0.38250 )
   2) LoyalCH < 0.48285 290  308.60 MM ( 0.22414 0.77586 )
     4) LoyalCH < 0.051325 55    0.00 MM ( 0.00000 1.00000 ) *
     5) LoyalCH > 0.051325 235  277.20 MM ( 0.27660 0.72340 )
```

```
      10) PriceDiff < 0.31 186  193.70 MM ( 0.21505 0.78495 ) *
      11) PriceDiff > 0.31 49   67.91 CH ( 0.51020 0.48980 ) *
   3) LoyalCH > 0.48285 510  446.50 CH ( 0.84118 0.15882 )
     6) LoyalCH < 0.764572 244  294.30 CH ( 0.70902 0.29098 )
      12) PriceDiff < 0.145 96  133.00 MM ( 0.48958 0.51042 )
        24) ListPriceDiff < 0.235 65   84.47 MM ( 0.35385 0.64615 ) *
        25) ListPriceDiff > 0.235 31   33.12 CH ( 0.77419 0.22581 ) *
      13) PriceDiff > 0.145 148  124.40 CH ( 0.85135 0.14865 ) *
     7) LoyalCH > 0.764572 266   85.24 CH ( 0.96241 0.03759 ) *
> plot(tree.oj, main="OJ Purchase Decision Tree")
> text(tree.oj, pretty=0)
> full.te.predict<-predict(tree.oj,oj.te, type="class")
> table(full.te.predict, oj.te$Purchase)

full.te.predict  CH  MM
             CH 132  27
             MM  27  84
> mean(full.te.predict!=oj.te$Purchase)
[1] 0.2
> cv.oj1=cv.tree(tree.oj)
> cv.oj1
$size
[1] 7 6 5 4 3 2 1

$dev
[1]  701.6317  701.2572  701.2572  711.0494  712.2034  778.9619 1067.5296

$k
[1]      -Inf  15.45135  15.60109  31.45203  36.81015  66.95200 309.35604

$method
[1] "deviance"

attr(,"class")
[1] "prune"         "tree.sequence"
> plot(cv.oj1$size,cv.oj1$dev,type="b", xlab="tree size",ylab="cross validated classificati
> prune.oj<-prune.misclass(tree.oj,best=5)
> plot(prune.oj)
> text(prune.oj, pretty=0)
> mean(predict(prune.oj,oj.tr,type="class")!=oj.tr$Purchase)
[1] 0.15875
```

```
> mean(predict(prune.oj, oj.te,type="class")!=oj.te$Purchase)
[1] 0.1962963
```