Group Assignment 7

Section 23 - Group 6 Project Groups (Bosan Hsu, Fan Liu, Jimeng Yin, Michael Liu, Richard Wang, Zhuoqian Zhang

2023-11-01

##a

```
library(ISLR)
library(tree)
attach(OJ)
set.seed(2)
train = sample(1:nrow(OJ), 800)
OJ.train=OJ[train,]
OJ.test=OJ[-train,]
```

##b

```
tree.oj <- tree(Purchase ~ ., data = OJ.train)
summary(tree.oj)

##
## Classification tree:
## tree(formula = Purchase ~ ., data = OJ.train)
## Variables actually used in tree construction:
## [1] "LoyalCH"   "PriceDiff"
## Number of terminal nodes:  9
## Residual mean deviance:  0.7009 = 554.4 / 791
## Misclassification error rate: 0.1588 = 127 / 800
```

##The error rate is 0.1588 and there are 9 nodes.The residual mean deviance is 0.7

##c

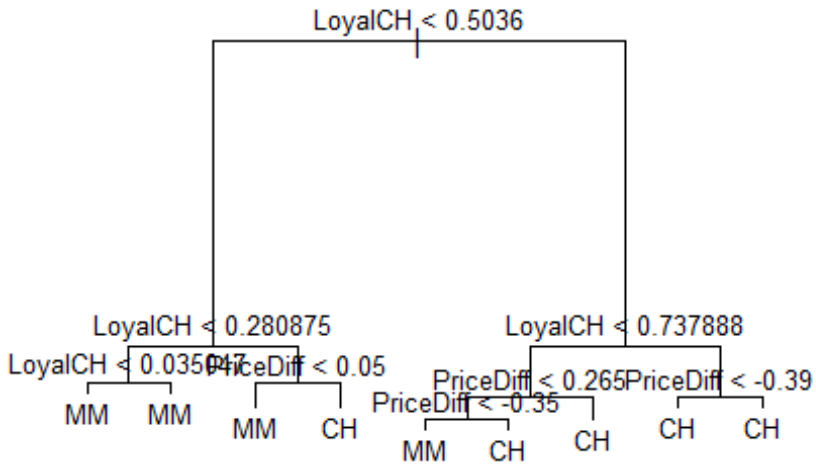
```
tree.oj

## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 800 1068.00 CH ( 0.61250 0.38750 )
##    2) LoyalCH < 0.5036 359  422.80 MM ( 0.27577 0.72423 )
##      4) LoyalCH < 0.280875 172  127.60 MM ( 0.12209 0.87791 )
##        8) LoyalCH < 0.035047 56   10.03 MM ( 0.01786 0.98214 ) *
##        9) LoyalCH > 0.035047 116  106.60 MM ( 0.17241 0.82759 ) *
##      5) LoyalCH > 0.280875 187  254.10 MM ( 0.41711 0.58289 )
##       10) PriceDiff < 0.05 73   71.36 MM ( 0.19178 0.80822 ) *
```

```
##       11) PriceDiff > 0.05 114  156.30 CH ( 0.56140 0.43860 ) *
##    3) LoyalCH > 0.5036 441  311.80 CH ( 0.88662 0.11338 )
##      6) LoyalCH < 0.737888 168  191.10 CH ( 0.74405 0.25595 )
##       12) PriceDiff < 0.265 93  125.00 CH ( 0.60215 0.39785 )
##         24) PriceDiff < -0.35 12   10.81 MM ( 0.16667 0.83333 ) *
##         25) PriceDiff > -0.35 81  103.10 CH ( 0.66667 0.33333 ) *
##       13) PriceDiff > 0.265 75   41.82 CH ( 0.92000 0.08000 ) *
##      7) LoyalCH > 0.737888 273   65.11 CH ( 0.97436 0.02564 )
##       14) PriceDiff < -0.39 11   12.89 CH ( 0.72727 0.27273 ) *
##       15) PriceDiff > -0.39 262   41.40 CH ( 0.98473 0.01527 ) *
```

##Node 8 is a significant node and there are 56 observations. It refers to that if the LoyalCH < 0.035047, then MM will be the choice

##d

```
plot(tree.oj)
text(tree.oj, pretty = 0, digits = 2, cex = 0.8)
```



##LoyalCH and PriceDiff are the most significant variables

##e

```
Purchase.test=Purchase[-train]
tree.pred=predict(tree.oj, OJ.test, type = "class")
table(tree.pred, Purchase.test)
```

```
##          Purchase.test
## tree.pred  CH  MM
##        CH 148  37
##        MM  15  70
```

```
mean(tree.pred!=Purchase.test)
```

```
## [1] 0.1925926
```

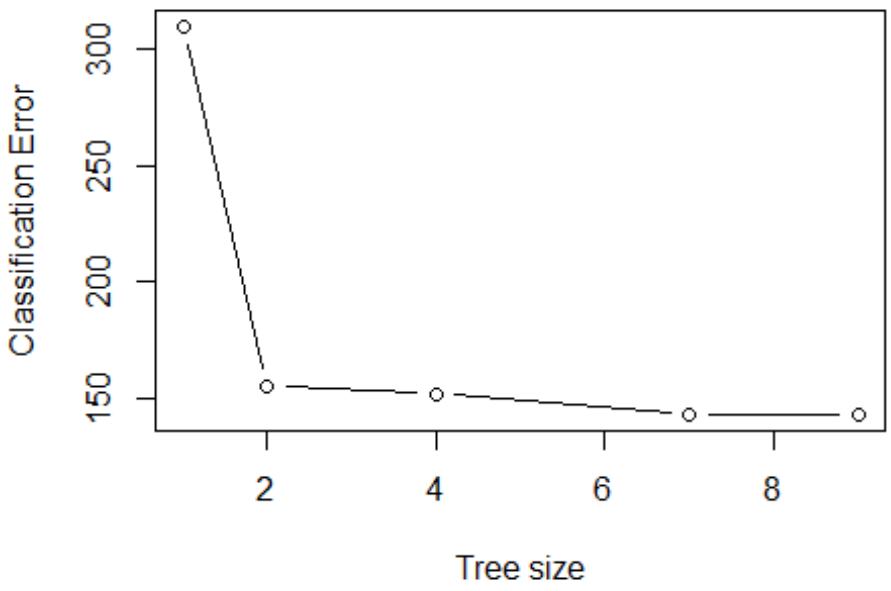
##f

```
cv.oj=cv.tree(tree.oj, FUN=prune.misclass)
cv.oj
```

```
## $size
## [1] 9 7 4 2 1
##
## $dev
## [1] 143 143 152 155 310
##
## $k
## [1]       -Inf   0.000000   2.666667   7.000000 161.000000
##
## $method
## [1] "misclass"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

##g

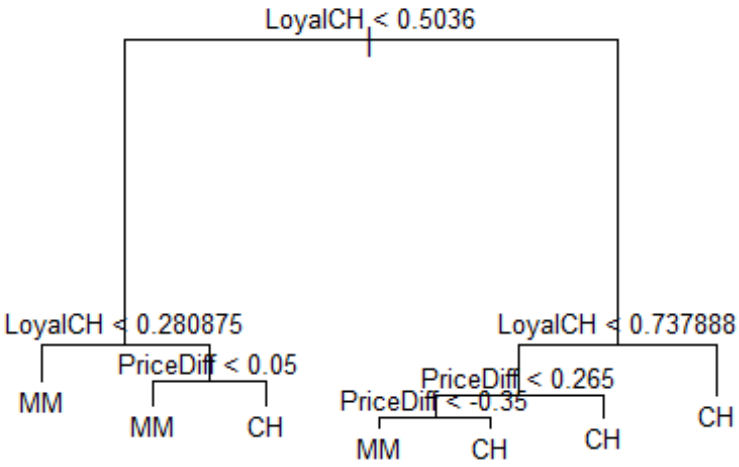
```
plot(cv.oj$size, cv.oj$dev, type = "b", xlab = "Tree size", ylab = "Cla
ssification Error")
```



##h ##Trees with 7 to 9 terminal nodes have the lowest Classification Errors

##i

```
prune.oj <- prune.misclass(tree.oj, best = 7)
plot(prune.oj)
text(prune.oj, pretty = 0, digits = 2, cex = 0.8)
```



##

##j

```
summary(prune.oj)
```

```
##
## Classification tree:
## snip.tree(tree = tree.oj, nodes = c(4L, 7L))
## Variables actually used in tree construction:
## [1] "LoyalCH"   "PriceDiff"
## Number of terminal nodes:  7
## Residual mean deviance:  0.7266 = 576.2 / 793
## Misclassification error rate: 0.1588 = 127 / 800
```

##we have the same Misclassification error rate

##k

```
tree.pred=predict(prune.oj,OJ.test,type="class")
table(tree.pred,Purchase.test)
```

```
##          Purchase.test
## tree.pred  CH  MM
##        CH 148  37
##        MM  15  70
```

```
mean(tree.pred!=Purchase.test)
```

```
## [1] 0.1925926
```

##The test error is the same