ST635 Homework 5 Solution

An Exercise on Classification Tree

1. Create a training set containing a random sample of 800 observations, and a test set containing the remaining observations. (10 points)
Solution:

```
library("ISLR")
id.train  = sample(1:1070,800,replace=FALSE)
data.train = OJ[id.train,]
data.test = OJ[-id.train,]
```

2. Fit a tree to the training data, with *Purchase* as the response and the other variables as predictors. Use the *summary()* function to produce summary statistics about the tree, and describe the results obtained. What is the training error rate? How many terminal nodes does the tree have? (10 points)
Solution: From the summary() output, there are 9 terminal nodes. The training misclassification error rate is 16.88%.

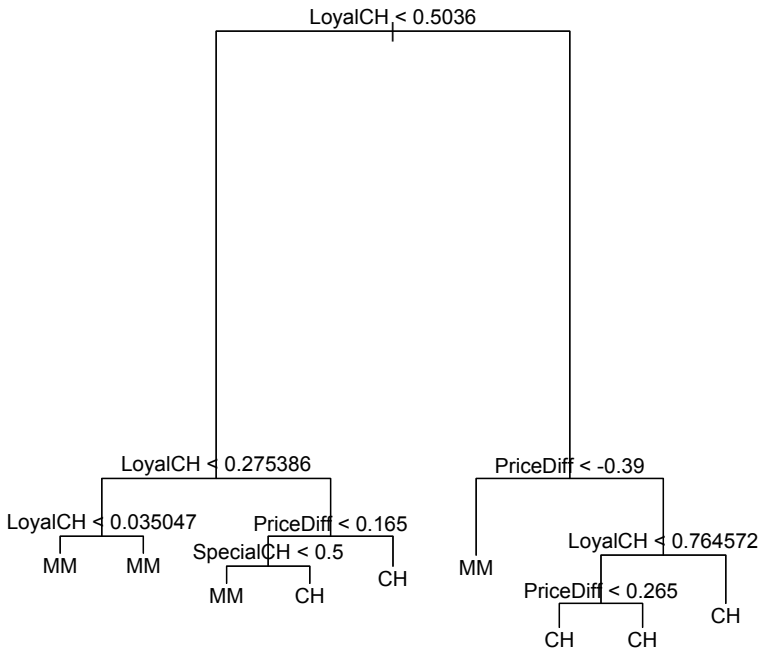
```
> library("tree")
> fit.tree = tree(Purchase~.,data=data.train)
> summary(fit.tree)
Classification tree:
tree(formula = Purchase ~ ., data = data.train)
Variables actually used in tree construction:
[1] "LoyalCH"   "PriceDiff" "SpecialCH"
Number of terminal nodes:  9
Residual mean deviance:  0.7607 = 601.7 / 791
Misclassification error rate: 0.1688 = 135 / 800
```

3. Type in the name of the tree object in order to get a detailed text output. Pick one of the terminal nodes, and interpret the information displayed. (10 points)
Solution: First note that only notes denoted as $*$ are terminal notes. For instance, nodes (2) and (4) in the output below are not terminal nodes. Suppose we pick node (8), the output indicates that all observations in this terminal node satisfy LoyalCH less than 0.035047; there are 56 observations in this region; the residual deviance of this node is 10.03; observations in this group are predicted to purchase Minutes Maid orange juice with probability 0.98214, and therefore the binary prediction for this terminal node is MM.

```
> fit.tree
node), split, n, deviance, yval, (yprob)
      * denotes terminal node
 1) root 800 1074.00 CH ( 0.60375 0.39625 )
   2) LoyalCH < 0.5036 347  407.40 MM ( 0.27378 0.72622 )
     4) LoyalCH < 0.275386 164  121.60 MM ( 0.12195 0.87805 )
       8) LoyalCH < 0.035047 56   10.03 MM ( 0.01786 0.98214 ) *
       9) LoyalCH > 0.035047 108  100.50 MM ( 0.17593 0.82407 ) *
     5) LoyalCH > 0.275386 183  247.70 MM ( 0.40984 0.59016 )
      10) PriceDiff < 0.165 79   84.79 MM ( 0.22785 0.77215 )
        20) SpecialCH < 0.5 66   56.14 MM ( 0.15152 0.84848 ) *
        21) SpecialCH > 0.5 13   17.32 CH ( 0.61538 0.38462 ) *
      11) PriceDiff > 0.165 104  143.20 CH ( 0.54808 0.45192 ) *
   3) LoyalCH > 0.5036 453  372.60 CH ( 0.85651 0.14349 )
     6) PriceDiff < -0.39 26   30.29 MM ( 0.26923 0.73077 ) *
     7) PriceDiff > -0.39 427  291.80 CH ( 0.89227 0.10773 )
      14) LoyalCH < 0.764572 170  175.50 CH ( 0.78824 0.21176 )
        28) PriceDiff < 0.265 94  117.70 CH ( 0.68085 0.31915 ) *
        29) PriceDiff > 0.265 76   41.98 CH ( 0.92105 0.07895 ) *
      15) LoyalCH > 0.764572 257   84.54 CH ( 0.96109 0.03891 ) *
```

4. Create a plot of the tree, and interpret the results. (20 points)
   Solution: Observations with LoyalCH$<$ 0.273586, observations with PriceDiff$<$ 0.165 and SpecialCH$<$ 0.5, Observations with PriceDiff$<$ -0.39 are predicted to purchase Minute Maid orange juice. Other observations are predicted to purchase CH orange juice. The detailed feature space for each terminal node can be expressed as follows.

```
> plot(fit.tree)
> text(fit.tree, pretty=0)
```



R_1 ={LoyalCH¡0.035}; observations in this region are more likely to purchase MM.
R_1 ={0.035¡= LoyalCH¡0.275 }; observations in this region are more likely to purchase MM.
R_3 ={Special CH¡0.5& Price Diff¡0.165& 0.275¡=Loyal CH¡0.504}; ; observations in this region are more likely to purchase MM.
R_4 ={Special CH ¿0.5& Price Diff¡0.165 & 0.275¡=Loyal CH¡0.504}; observations in this region are more likely to purchase CH.
R_5 ={Price Diff¿0.165& 0.275¡=Loyal CH¡0.504}; observations in this region are more likely to purchase CH.
R_6 ={Price Diff¡0.39& Loyal CH¿=0.504}; observations in this region are more likely to purchase MM.
R_7 ={-0.39¡=Price Diff¡0.265& 0.504¡= Loyal CH¡0.765};observations in this region are more likely to purchase CH.
R_8 ={Price Diff¿=0.265& 0.504¡= Loyal CH¡0.765};observations in this region are more likely to purchase CH.
R_9 ={Price Diff¿=-0.39& Loyal CH¿=0.765};observations in this region are more likely to purchase CH.

5. Predict the response on the test data. What is the test error rate? (10 points)
Solution: The test error rate is $(28 + 17)/270 = 16.67\%$.

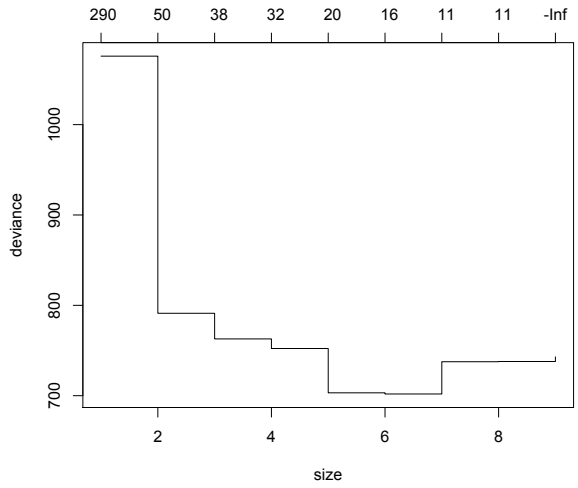
```
> test.pred = predict(fit.tree,data.test)
> test.Y = apply(test.pred,1,which.max)
> test.Y[test.Y==1]="CH"
> test.Y[test.Y==2]="MM"
>
> table(test.Y,data.test$Purchase)
test.Y  CH  MM
    CH 153  28
    MM  17  72
```

Note: You may also obtain the binary prediction directly by using

```
predict(fit.tree,data.test,type="class")
```

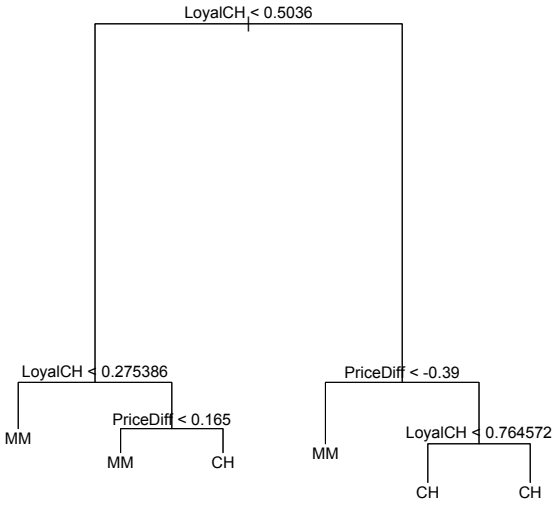
6. Apply the *cv.tree()* function to the training set in order to determine the optimal tree size using ten-fold cross-validation. Produce a plot with tree size on the horizontal axis and cross-validated classification error rate on the vertical axis. Which tree size corresponds to the lowest cross-validated classification error rate? (10 points)
Solution: A tree with six terminal nodes seems to have the smallest cross-validated error rate.

```
> result = cv.tree(fit.tree,K=10,FUN=prune.tree)
> plot(result)
```



7. Produce a pruned tree corresponding to the optimal tree size obtained using cross-validation. If cross-validation does not lead to selection of a pruned tree, then create a pruned tree with five terminal nodes. (10 points)
Solution:

```
> fit.prune = prune.tree(fit.tree,best=6)
> plot(fit.prune)
> text(fit.prune,pretty=0)
```



8. Compare the training error rates between the pruned and un-pruned trees. Which is higher? (10 points)
Solution: The training error rate based on the pruned tree is 17.5%, which is slightly higher than that based on the un-pruned tree.

```
> summary(fit.prune)
Classification tree:
snip.tree(tree = fit.tree, nodes = 4:5)
Variables actually used in tree construction:
[1] "LoyalCH"       "PriceDiff"     "ListPriceDiff"
Number of terminal nodes:  6
Residual mean deviance:  0.809 = 642.3 / 794
Misclassification error rate: 0.175 = 140 / 800
```

9. Compare the test error rates between the pruned and un-pruned trees. Which is higher? (10 points)
Solution: The test misclassification rate based on the pruned tree is 15.19%, which is lower than that based on the un-pruned tree.

```
> test.pred2 = predict(fit.prune,data.test)
> test.Y2 = apply(test.pred2,1,which.max)
> test.Y2[test.Y2==1]="CH"
> test.Y2[test.Y2==2]="MM"
> table(test.Y2,data.test$Purchase)
test.Y2  CH  MM
     CH 129  12
     MM  29 100
> 41/270
[1] 0.1518519
```

Note: Your results may be slightly different due to the random generation of the training and test sets.