HW9
(Roger)Siwei Zhang
20335254

Chapter 8: Exercise 9

a.

```
library(ISLR)
attach(OJ)
set.seed(1)
head(OJ)
dim(OJ)
train = sample(dim(OJ)[1],800)
OJ_train = OJ[train,]
OJ_test  = OJ[-train,]
```

b.

```
library(tree)
oj_tree = tree(Purchase~.,data=OJ_train)
summary(oj_tree)
```

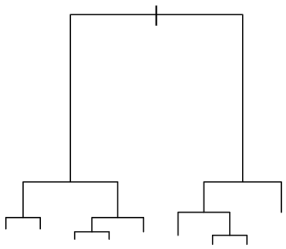
```
## Number of terminal nodes:  7
## Residual mean deviance:  0.752 = 596 / 793
## Misclassification error rate: 0.155 = 124 / 800
```

c.

```
> oj_tree
node), split, n, deviance, yval, (yprob)
      * denotes terminal node
 1) root 800 1070.000 CH ( 0.61000 0.39000 )
 2) LoyalCH < 0.5036 346  419.600 MM ( 0.29480 0.70520 )
 4) LoyalCH < 0.275354 154   98.360 MM ( 0.09740 0.90260 )
 8) LoyalCH < 0.064156 66    0.000 MM ( 0.00000 1.00000 ) *
 9) LoyalCH > 0.064156 88   80.360 MM ( 0.17045 0.82955 ) *
 5) LoyalCH > 0.275354 192  264.500 MM ( 0.45312 0.54688 )
 10) SalePriceMM < 2.04 102  123.600 MM ( 0.29412 0.70588 )
 20) SpecialCH < 0.5 78   79.160 MM ( 0.20513 0.79487 ) *
 21) SpecialCH > 0.5 24   32.600 CH ( 0.58333 0.41667 ) *
 11) SalePriceMM > 2.04 90  118.300 CH ( 0.63333 0.36667 ) *
 3) LoyalCH > 0.5036 454  383.500 CH ( 0.85022 0.14978 )
 6) LoyalCH < 0.753545 188  225.500 CH ( 0.71277 0.28723 )
 12) PriceDiff < -0.35 17    7.606 MM ( 0.05882 0.94118 ) *
 13) PriceDiff > -0.35 171  181.200 CH ( 0.77778 0.22222 )
  26) SalePriceMM < 2.125 106  131.500 CH ( 0.68868 0.31132 ) *
  27) SalePriceMM > 2.125 65   35.250 CH ( 0.92308 0.07692 ) *
  7) LoyalCH > 0.753545 266  109.700 CH ( 0.94737 0.05263 ) *
```

For example, 2). The variable is LoyalCH with value 0.50. There are 364 subtree points below this node. The points deviance in this region is 419.6. The prediction of this node is 0.29~0.71.

d.
plot(oj_tree)



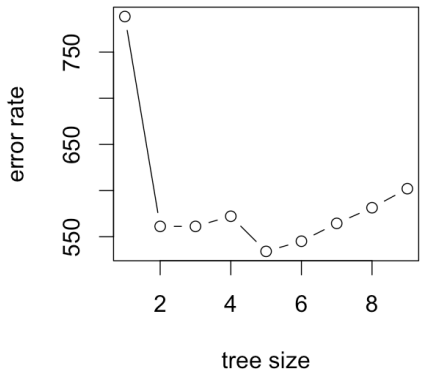
Didn’t see the result in the tree, try to solve this problem later.

e.
oj_pred = predict(oj_tree, OJ_test, type = "class")
table(OJ_test$Purchase, oj_pred)

```
    oj_pred
    CH  MM
 CH 146  13
 MM  32  79
```

f.
cv_oj = cv.tree(oj_tree, FUN = prune.tree)

g.
plot(cv_oj$size, cv_oj$dev, type = "b", xlab = "tree size", ylab = "error rate”)



h.
Size 5 gives lowest cross-validated classification error rate.

i.
oj_pruned = prune_tree(oj_tree, best = 6)

j.
summary(oj_pruned)
Pruned higher.

k.

```
pred_unpruned = predict(oj_tree, OJ_test, type = "class")
misclass_unpruned = sum(OJ_test$Purchase != pred_unpruned)
misclass_unpruned/length(pred_unpruned)

pred_pruned = predict(oj_pruned, OJ_test, type = "class")
misclass_pruned = sum(OJ_test$Purchase != pred_pruned)
misclass_pruned/length(pred_pruned)
```

[1] 0.1666667 unpruned
[1] 0.1592593 pruned

Unpruned is higher.