Individual assignment 7

#Student ID:474084, Yiqing Zhang

```
library(ISLR)
attach(Carseats)
```

#a

```
set.seed(100)
train=sample(1:nrow(Carseats),nrow(Carseats)/2)
car.train=Carseats[train,]
car.test=Carseats[-train,]
```

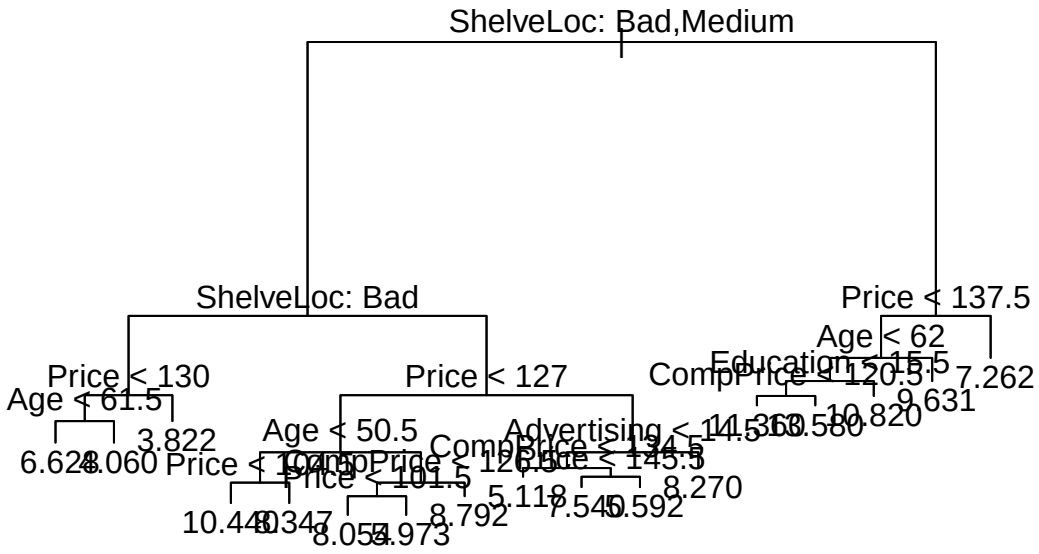
In this chunck, I split the data set into two part by the sample function. And split it to train and test data according to the row number of Carseats.

#b

```
library(tree)
set.seed(100)
car.tree=tree(Sales~.,car.train)
car.pred=predict(car.tree,car.test)
summary(car.tree)
```

```
##
## Regression tree:
## tree(formula = Sales ~ ., data = car.train)
## Variables actually used in tree construction:
## [1] "ShelveLoc"   "Price"       "Age"         "CompPrice"   "Advertising"
## [6] "Education"
## Number of terminal nodes:  17
## Residual mean deviance:  1.844 = 337.5 / 183
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.81700 -0.89580 -0.01857  0.00000  0.92000  2.91800
```

```
plot(car.tree)
text(car.tree,pretty=0)
```




```
tmse=mean((car.test$Sales-car.pred)^2)
tmse
```

```
## [1] 5.395751
```

In this chunck, the tree model uses 6 predictors and 17 nodes. The Test MSE is 5.395751. For example,in this plot, if the Shelveloc is good and price is bigger than or equal to 137.5, the mean of sales would be 7.262 according to the tree model.

#c

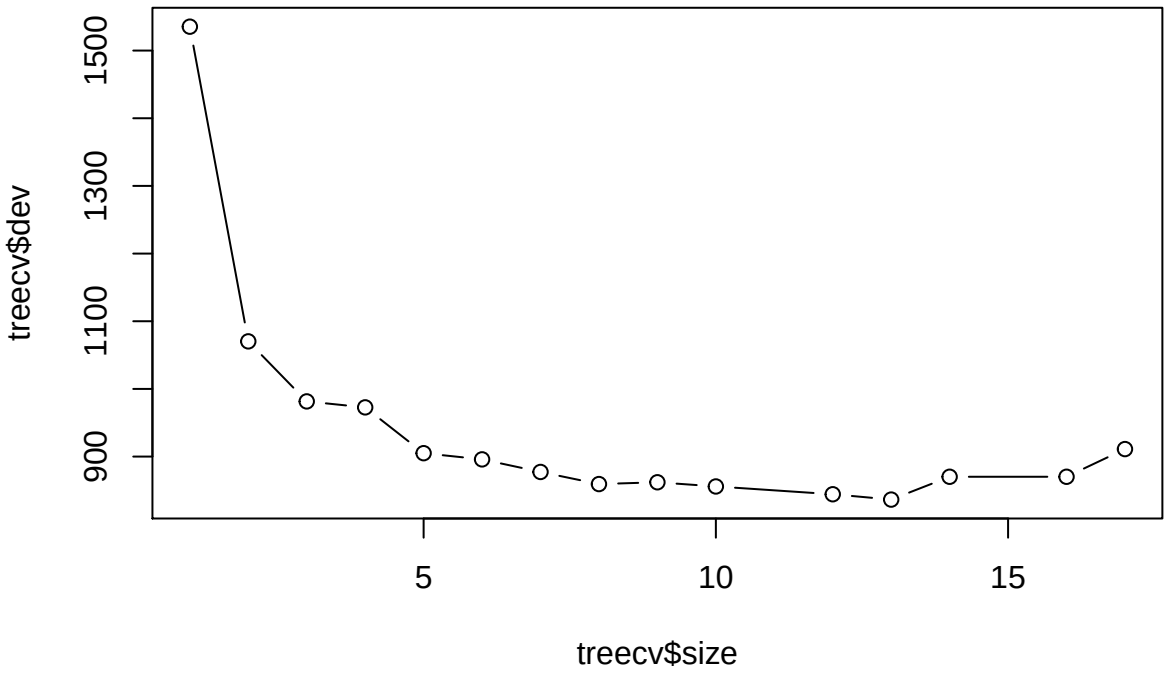
```
set.seed(100)
treecv=cv.tree(car.tree)
treecv
```

```
## $size
##  [1] 17 16 14 13 12 10  9  8  7  6  5  4  3  2  1
##
## $dev
##  [1]  911.0873  870.1659  870.1659  836.4578  844.3387  855.8490  862.0321
##  [8]  859.3948  877.2921  895.8563  905.0300  972.7206  981.5440 1070.2611
## [15] 1535.3160
##
## $k
##  [1]      -Inf  16.29507  16.51993  25.79705  27.26126  27.92693  35.37058
##  [8]  37.18435  40.02062  44.83473  52.50480  72.69219  98.74993 137.17898
## [15] 473.72357
##
## $method
## [1] "deviance"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

```
which.min(treecv$dev)
```

```
## [1] 4
```

```
plot(treecv$size,treecv$dev,type="b")
```




```
car.prune=prune.tree(car.tree,best=13)
ycar=predict(car.prune,car.test)
mean((car.test$Sales-ycar)^2)
```

```
## [1] 5.39241
```

In this chunck, I use cross validation to decide that the optimal nodes of this tree model are 13, which has the least training MSE. And the pruned model's Test MSE is 5.39241. And the Test MSE of this pruned model is slightly better than original model.