ch8

2 It is mentioned boosting using depth-one trees lead to an additive model

$f(x) = \sum_{j=1}^{p} f_j(X_j)$ Explain why using

$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x)$ => Setting $\hat{f}(x) = 0$ $r_i = y_i$ for all i in training set

$\downarrow$

$f_1(x_1) = \lambda \hat{f}^1(x)$

=> $\hat{f}^1(x) = \frac{f_1(x_1)}{\lambda}$ => $\lambda \hat{f}^1(x) = f(x)$

$r_i \leftarrow r_i - \lambda \hat{f}^b(x_i)$

$r_1 = y_1 - \lambda \hat{f}^1(x_1)$

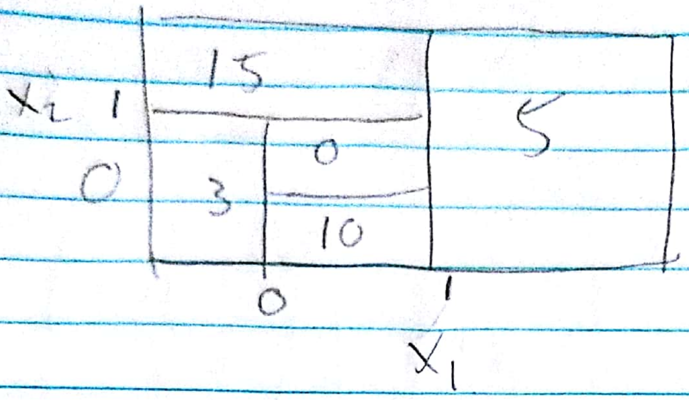
$\hat{f}^2(x) = \frac{1}{\lambda} f_2(x_2)$

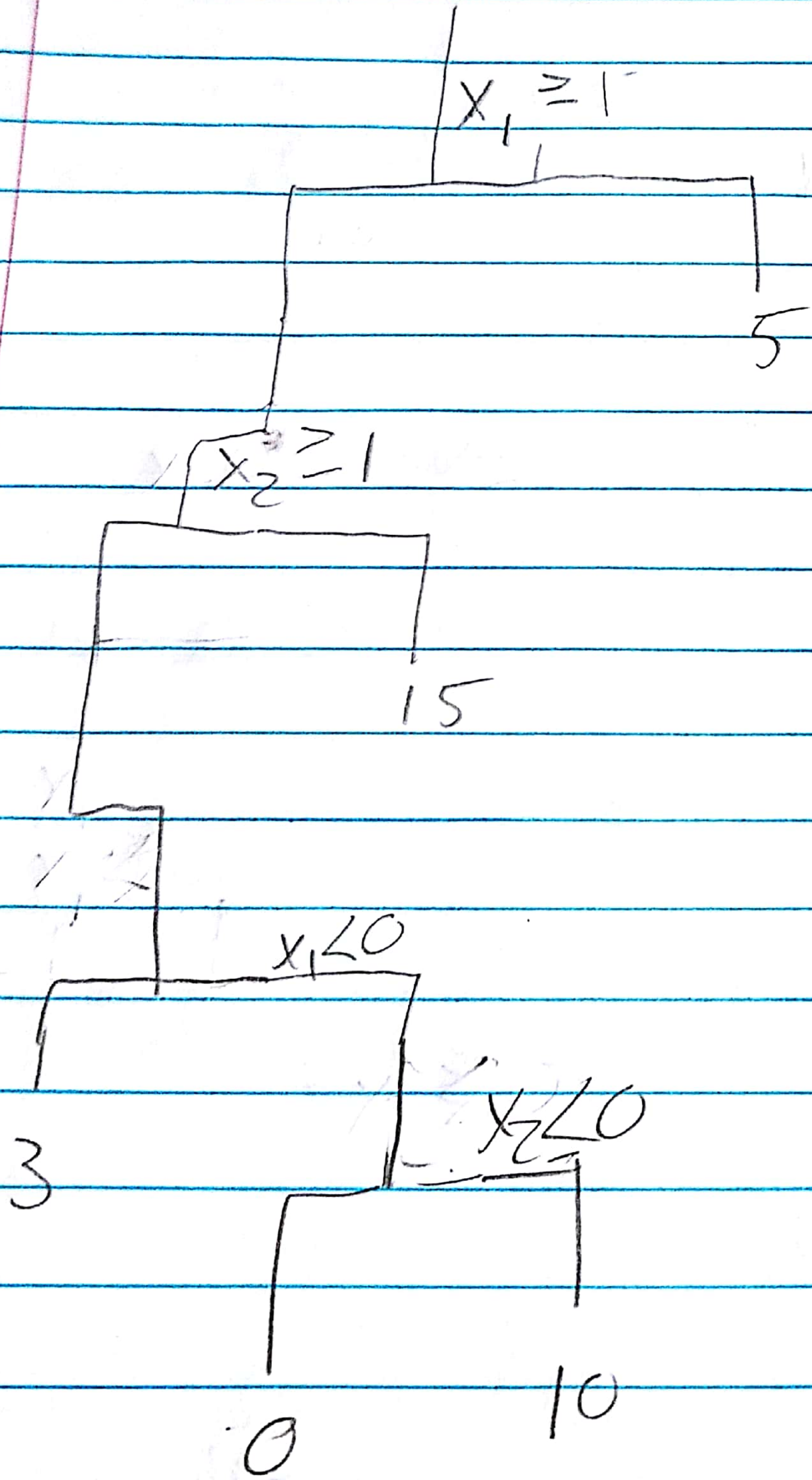
$f(x) = \lambda \hat{f}^1(x) + \lambda \hat{f}^2(X)$ from 8.12

$r_i = y_i - \lambda \hat{f}^1(x) - \lambda \hat{f}^2(x)$

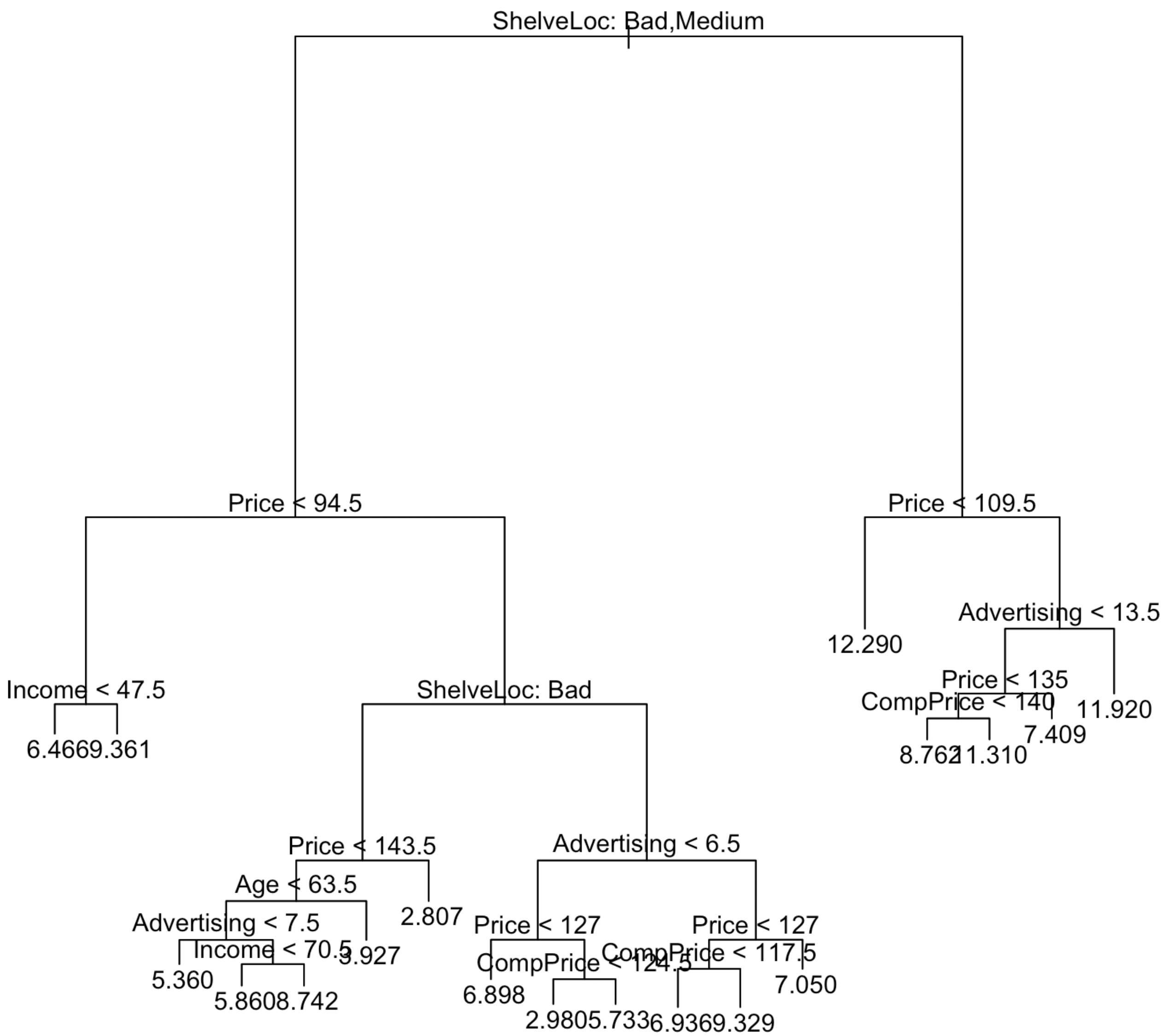
which $\hat{f}(x) = \sum_{j=1}^{p} f_j(x_j)$

4a

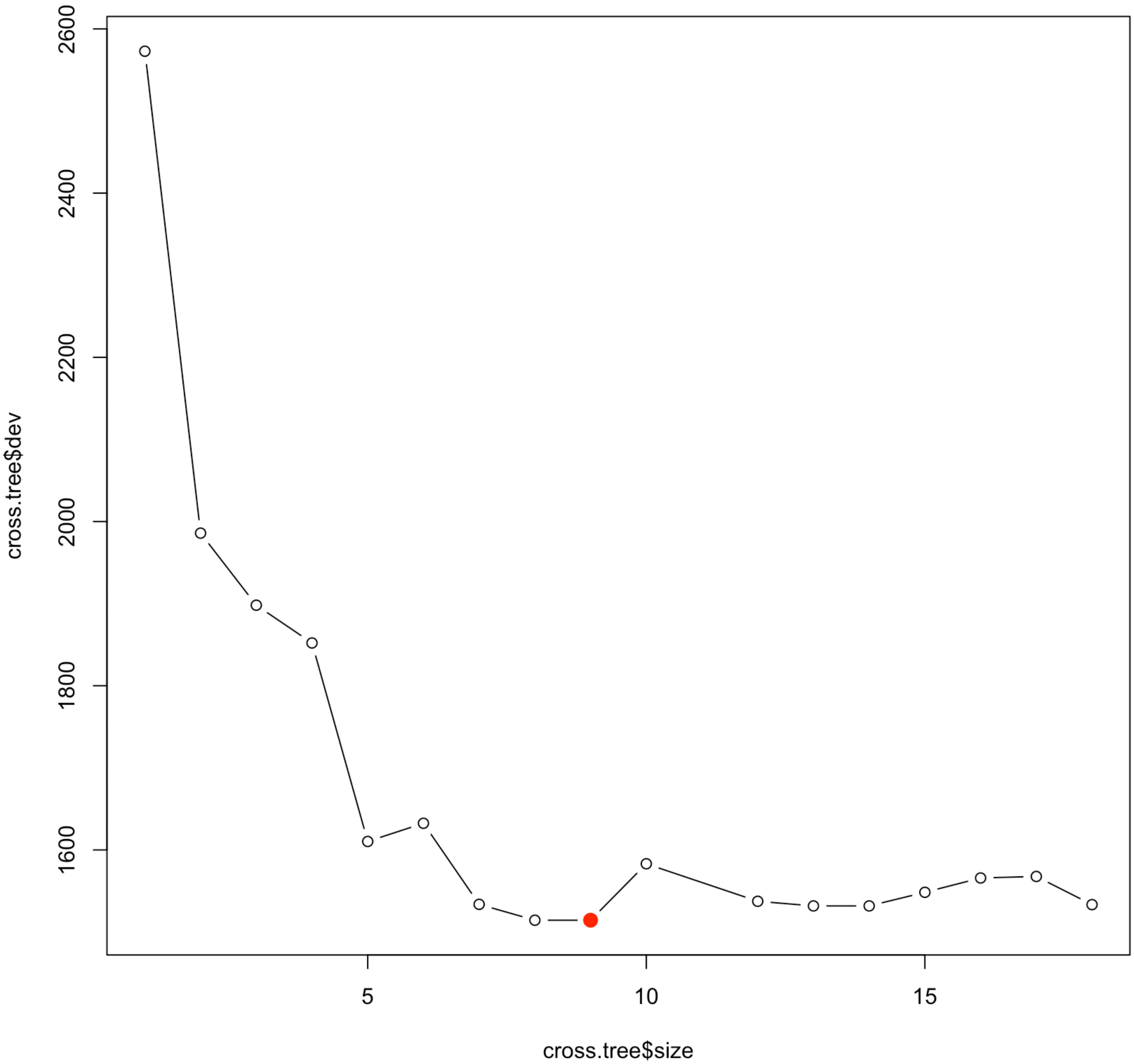




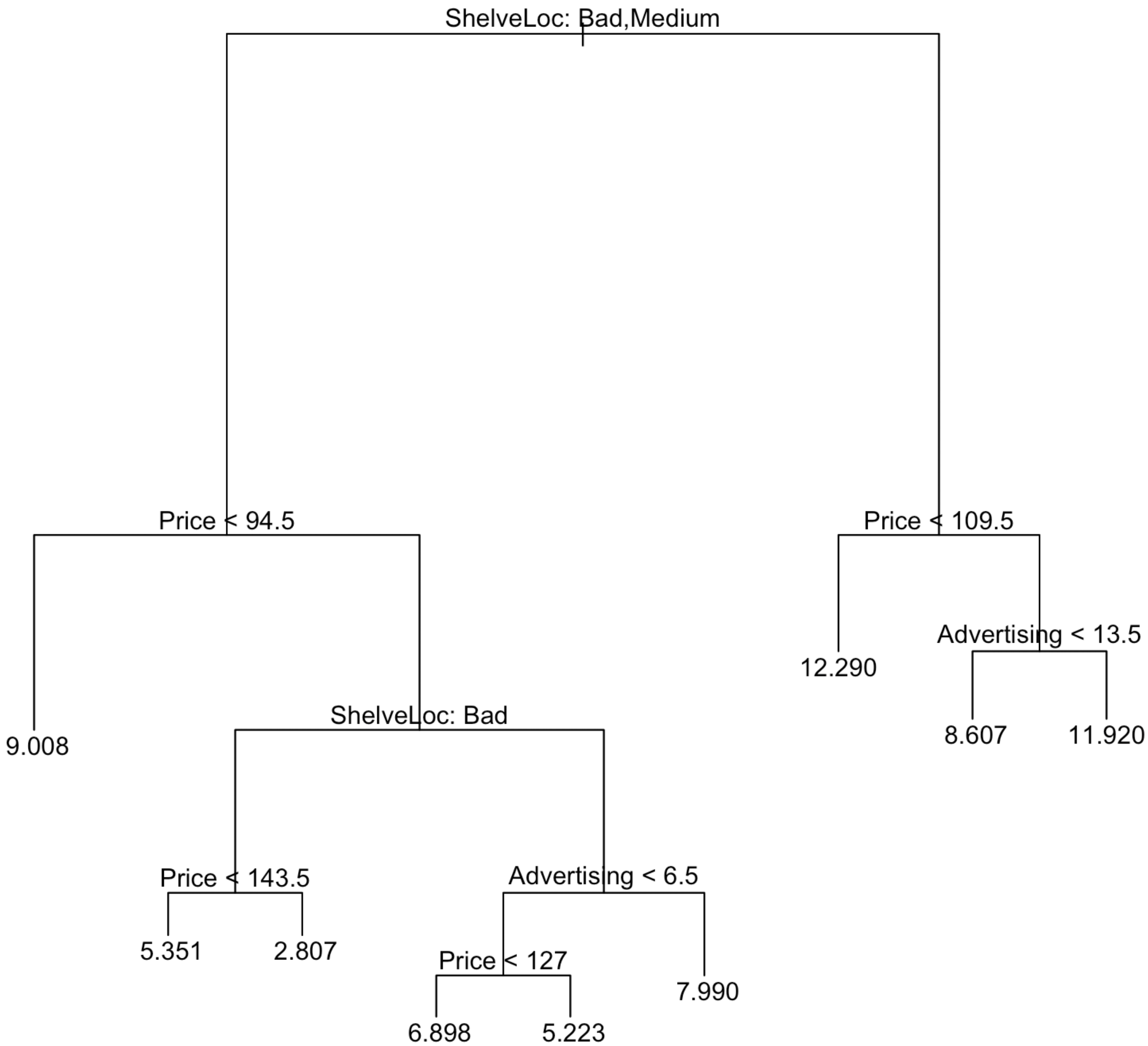
```
library(ISLR)

data = Carseats

head(data)
#part A
set.seed(333)
samp <-sample(1:nrow(data), floor(0.8*nrow(data)))

training <- data[samp,]



test <- data[-samp,]

#Part B

library(tree)
tree.model <- tree(Sales ~ ., data = training)
summary(tree.model)

plot(tree.model)
text(tree.model, pretty = 0)

ypred = predict(tree.model, newdata = test)
mean((ypred - test$Sales)^2)



[1] 3.432136
```




```
#Part C
set.seed(333)
cross.tree<- cv.tree(tree.model)
plot(cross.tree$size,cross.tree$dev,type = 'b')
tree.min = which.min(cross.tree$dev)
points(tree.min,cross.tree$dev[tree.min],col = 'red', cex = 2 , pch = 20)
```




```
prune.tree <- prune.tree(cross.tree , best =9)
plot(prune.tree)
text(prune.tree , pretty = 0)
```

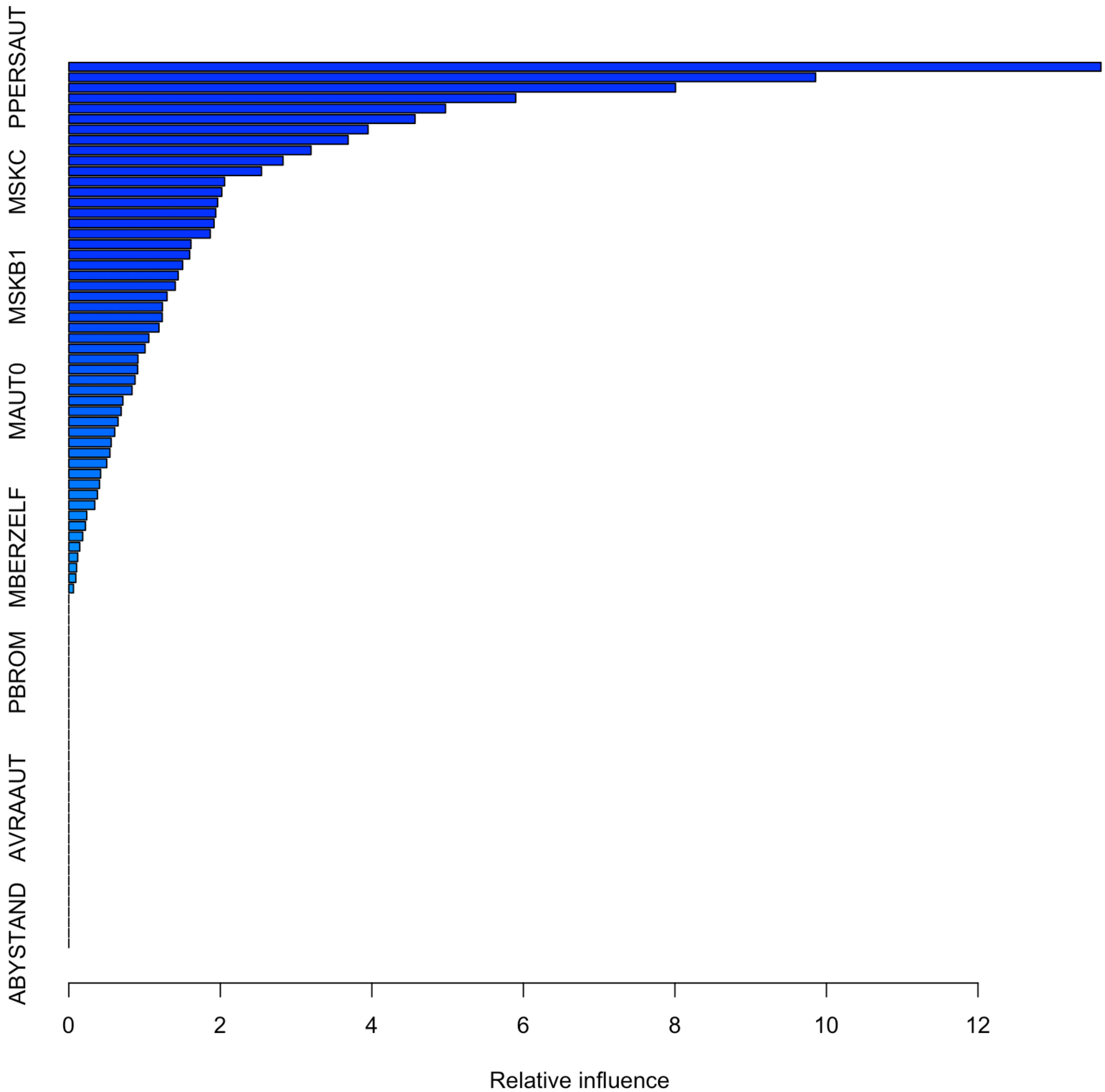



```
ypred = predict(prune.tree, newdata = test)
mean((ypred - test$Sales)^2)
[1] 4.283055



library(randomForest)
bag <-  randomForest(Sales ~.,data = training , mtry = 10 , ntree = 500, impor
tance = T)
ypred <- predict(bag, newdata = test)
```

```
mean(( ypred - test$Sales)^2)
[1] 1.916549
importance(bag)
> importance(bag)
               %IncMSE IncNodePurity
CompPrice   29.6469632     231.26134
Income       6.5266834     113.69830
Advertising 24.4109171     209.83447
Population  -0.8263424      90.11008
Price       68.6482009     719.84665
ShelveLoc   76.5874359     795.16790
Age         20.0234367     236.13359
Education    1.8579601      68.25215
Urban       -3.0802310      10.89329
US           1.9182432      12.45439
rf <- randomForest(Sales~.,data = training, mtry = 3 , ntree = 500 , importanc
e = T)
rf.pred <- predict(rf, newdata= test)
mean((rf.pred-test$Sales)^2)
[1] 2.696763
importance(rf)
> importance(rf)
              %IncMSE IncNodePurity
CompPrice   12.160208     206.44868
Income       2.655173     172.70728
Advertising 19.078848     232.54135
Population  -0.946113     162.87398
Price       41.615778     581.20243
ShelveLoc   51.051150     599.33333
Age         15.932367     270.81909
Education    3.301610     108.67523
Urban       -2.033248      19.52928
US           3.391320      35.10130
```

Exercise 11

```
#part a
library(gbm)
data = Caravan

head(data)
set.seed(333)
trains <- 1:1000
#part b
Caravan$Purchase <- ifelse(Caravan$Purchase == "Yes", 1, 0)
c.train <- Caravan[trains,]
c.test <- Caravan[-trains,]

set.seed(333)

boost.c <- gbm(Purchase ~.,data = c.train,distribution = "gaussian", n.trees =
 1000, shrinkage = 0.01)
xlim = x
summary(boost.c)
```




```
> summary(boost.c)
              var     rel.inf
PPERSAUT PPERSAUT 13.62317704
MKOOPKLA MKOOPKLA  9.85725691
MOPLHOOG MOPLHOOG  8.00855861
MBERMIDD MBERMIDD  5.89901037
ABRAND     ABRAND  4.97233525
PBRAND     PBRAND  4.57062957
```

MGODGE	MGODGE	3.95052868
MINK3045	MINK3045	3.68743021
PWAPART	PWAPART	3.19752944
MAUT1	MAUT1	2.82826871
MOSTYPE	MOSTYPE	2.54426049
MSKC	MSKC	2.05673676
MBERHOOG	MBERHOOG	2.01991873
MAUT2	MAUT2	1.96485617
MGODPR	MGODPR	1.94012211
MSKA	MSKA	1.91753653
PBYSTAND	PBYSTAND	1.86746832
MBERARBG	MBERARBG	1.61280273
MRELGE	MRELGE	1.59545525
MGODOV	MGODOV	1.50295598
MFGEKIND	MFGEKIND	1.44410397
MSKB1	MSKB1	1.40561281
MFWEKIND	MFWEKIND	1.29723503
MOPLMIDD	MOPLMIDD	1.23593826
MINK7512	MINK7512	1.23405415
MGODRK	MGODRK	1.19020178
MOSHOOFD	MOSHOOFD	1.05666306
MINKGEM	MINKGEM	1.00708231
MZFONDS	MZFONDS	0.91299252
MHHUUR	MHHUUR	0.90937952
MINK4575	MINK4575	0.87490430
MINKM30	MINKM30	0.83519801
MAUT0	MAUT0	0.71461301
MGEMOMV	MGEMOMV	0.69189049
MBERBOER	MBERBOER	0.65053573
MBERARBO	MBERARBO	0.60705381
MHKOOP	MHKOOP	0.55995757
MRELOV	MRELOV	0.54346751
MGEMLEEF	MGEMLEEF	0.50103082
MSKD	MSKD	0.42050029
MINK123M	MINK123M	0.40557816
MSKB2	MSKB2	0.37807134
PMOTSCO	PMOTSCO	0.34349082

MZPART	MZPART	0.23712117
MOPLLAAG	MOPLLAAG	0.22029231
MRELSA	MRELSA	0.18411121
MBERZELF	MBERZELF	0.14536101
APERSAUT	APERSAUT	0.11818064
PLEVEN	PLEVEN	0.10334514
MAANTHUI	MAANTHUI	0.09317822
MFALLEEN	MFALLEEN	0.06201720
PWABEDR	PWABEDR	0.00000000
PWALAND	PWALAND	0.00000000
PBESAUT	PBESAUT	0.00000000
PVRAAUT	PVRAAUT	0.00000000
PAANHANG	PAANHANG	0.00000000
PTRACTOR	PTRACTOR	0.00000000
PWERKT	PWERKT	0.00000000
PBROM	PBROM	0.00000000
PPERSONG	PPERSONG	0.00000000
PGEZONG	PGEZONG	0.00000000
PWAOREG	PWAOREG	0.00000000
PZEILPL	PZEILPL	0.00000000
PPLEZIER	PPLEZIER	0.00000000
PFIETS	PFIETS	0.00000000
PINBOED	PINBOED	0.00000000
AWAPART	AWAPART	0.00000000
AWABEDR	AWABEDR	0.00000000
AWALAND	AWALAND	0.00000000
ABESAUT	ABESAUT	0.00000000
AMOTSCO	AMOTSCO	0.00000000
AVRAAUT	AVRAAUT	0.00000000
AAANHANG	AAANHANG	0.00000000
ATRACTOR	ATRACTOR	0.00000000
AWERKT	AWERKT	0.00000000
ABROM	ABROM	0.00000000
ALEVEN	ALEVEN	0.00000000
APERSONG	APERSONG	0.00000000
AGEZONG	AGEZONG	0.00000000
AWAOREG	AWAOREG	0.00000000

```
AZEILPL   AZEILPL  0.00000000
APLEZIER APLEZIER  0.00000000
AFIETS     AFIETS  0.00000000
AINBOED   AINBOED  0.00000000
ABYSTAND ABYSTAND  0.00000000
>
```