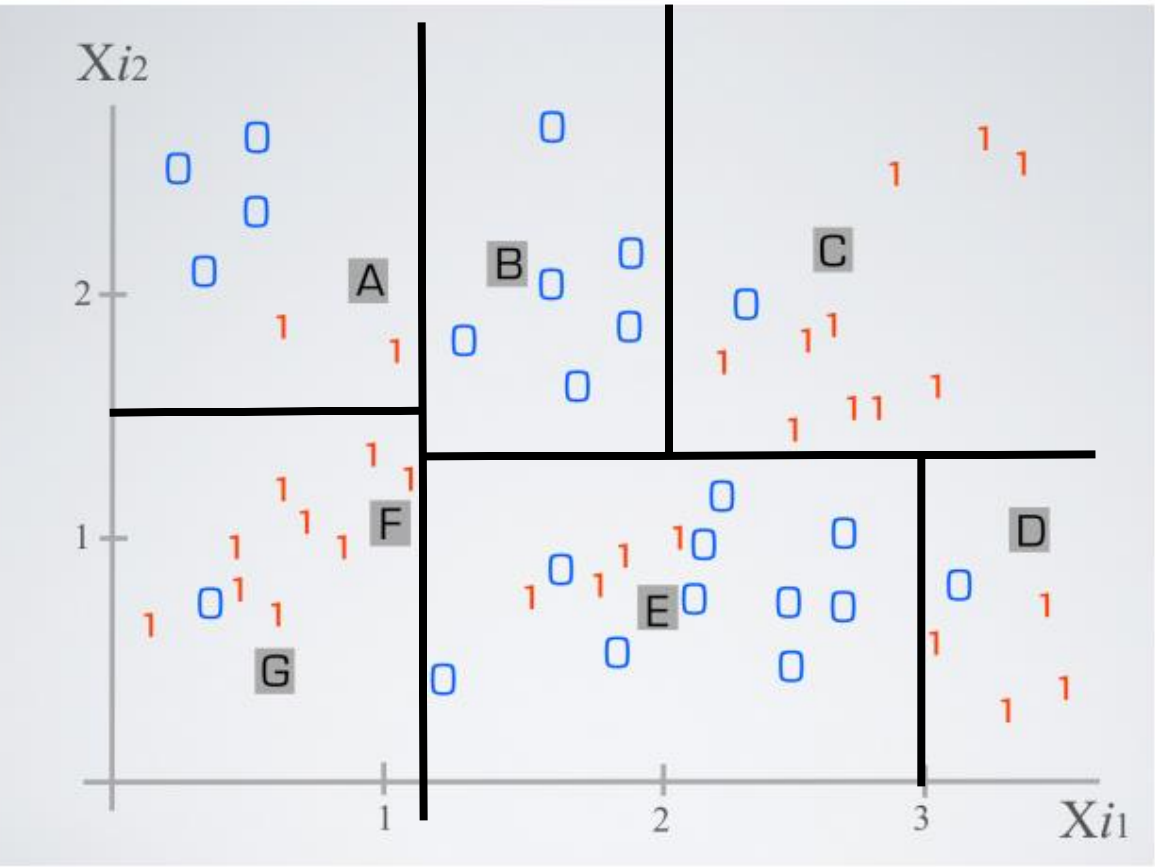
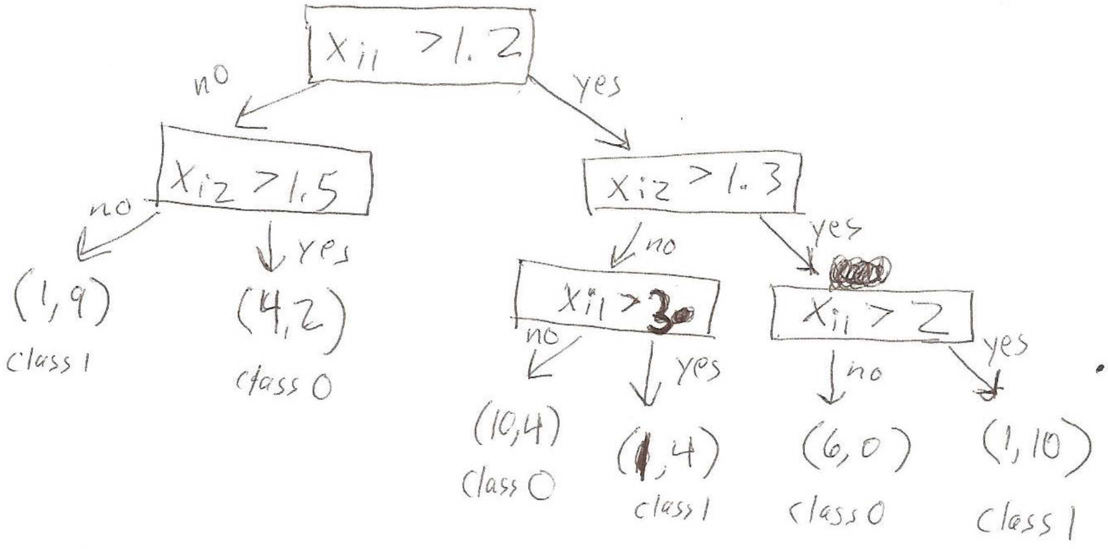
## Problem 1: The Correspondence between Data and Trees





Note: I used the majority rule in determing the classes

Point A: Class 0

Point B: Class 0

Point C: Class 1

Point D: Class 1

Point E: Class 0

Point F: Class 1

Point G: Class 1

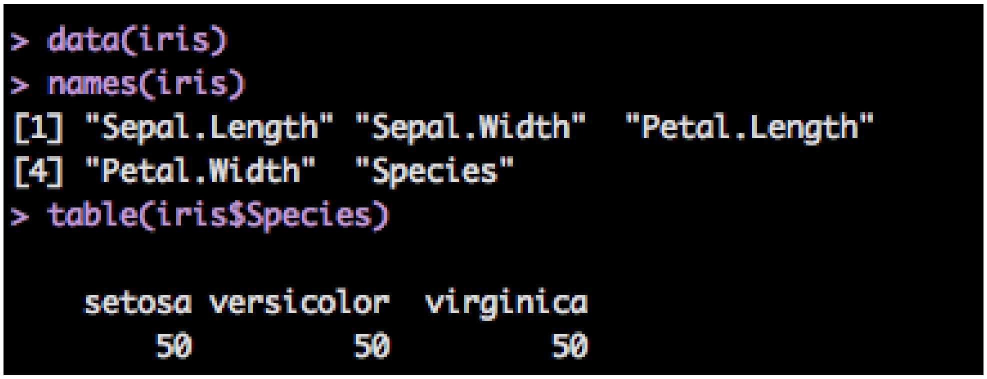
## Problem 2: Trees in R

### Part A: Iris Data Set

> data(iris)

> names(iris)

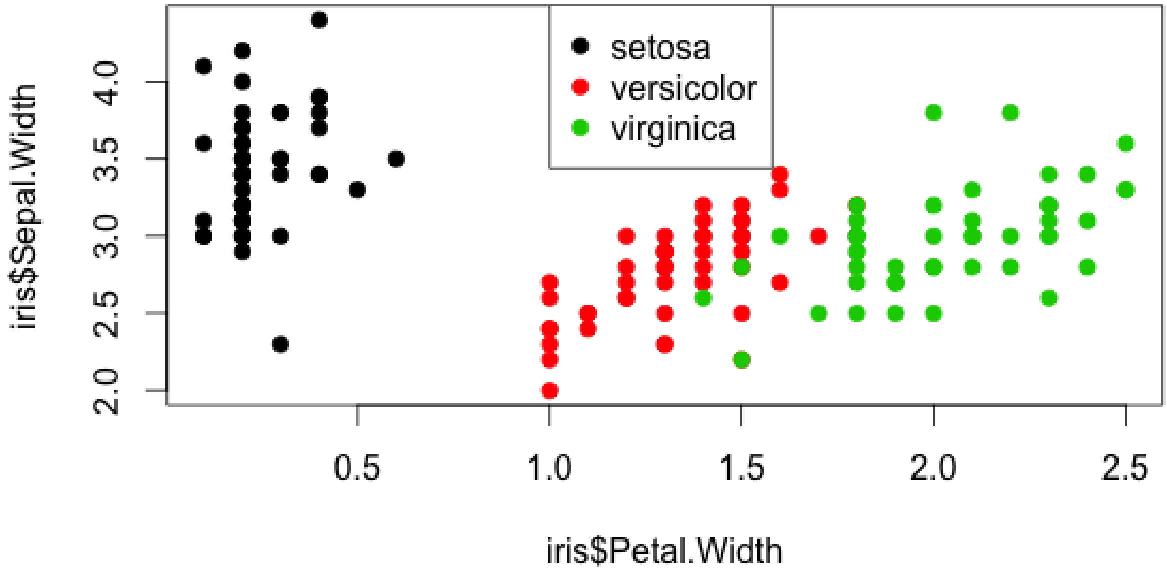
> table(iris$Species)



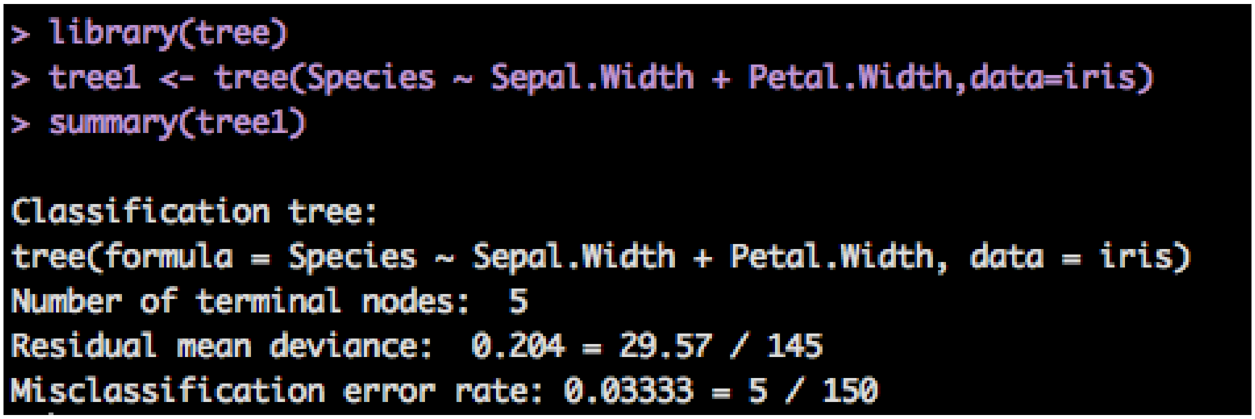
```
> data(iris)
> names(iris)
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length"
[4] "Petal.Width"  "Species"
> table(iris$Species)

    setosa versicolor  virginica
        50         50         50
```

> plot(iris$Petal.Width, iris$Sepal.Width, pch = 19, col = as.numeric(iris$Species))

> legend(1, 4.5, legend = unique(iris$Species), col = unique(as.numeric(iris$Species)), pch = 19)

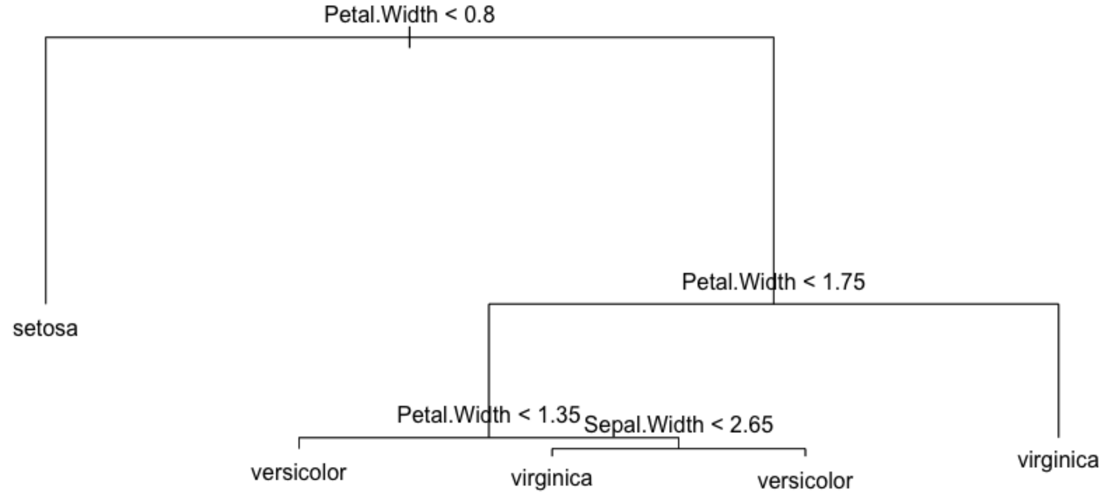


```
> install.packages("tree")
> library(tree)
> tree1 <- tree(Species ~ Sepal.Width + Petal.Width, data = iris)
> summary(tree1)
```

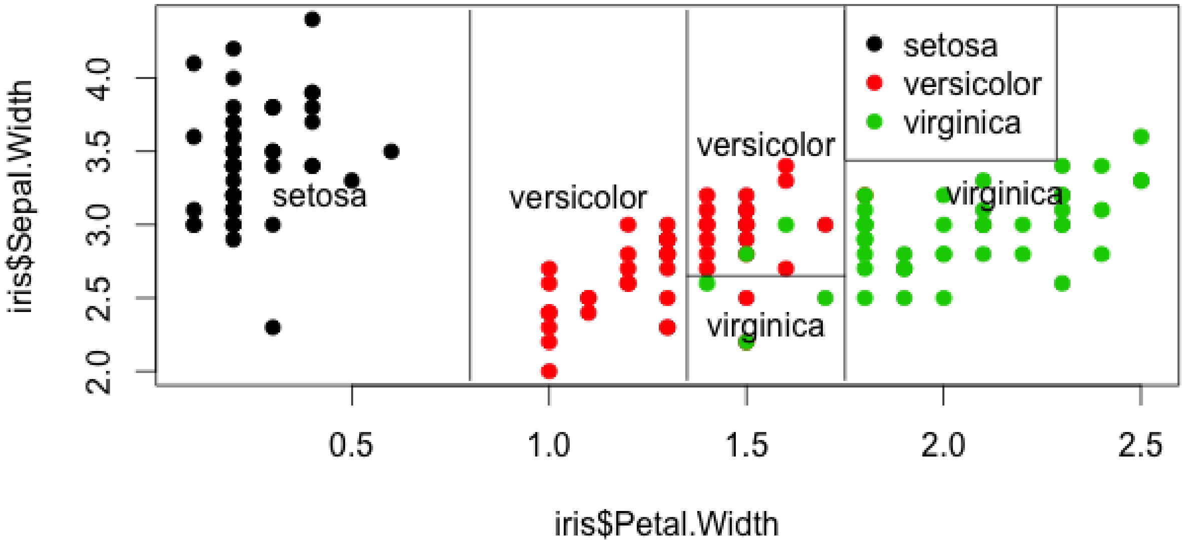


```
> library(tree)
> tree1 <- tree(Species ~ Sepal.Width + Petal.Width,data=iris)
> summary(tree1)

Classification tree:
tree(formula = Species ~ Sepal.Width + Petal.Width, data = iris)
Number of terminal nodes:  5
Residual mean deviance:  0.204 = 29.57 / 145
Misclassification error rate: 0.03333 = 5 / 150
```

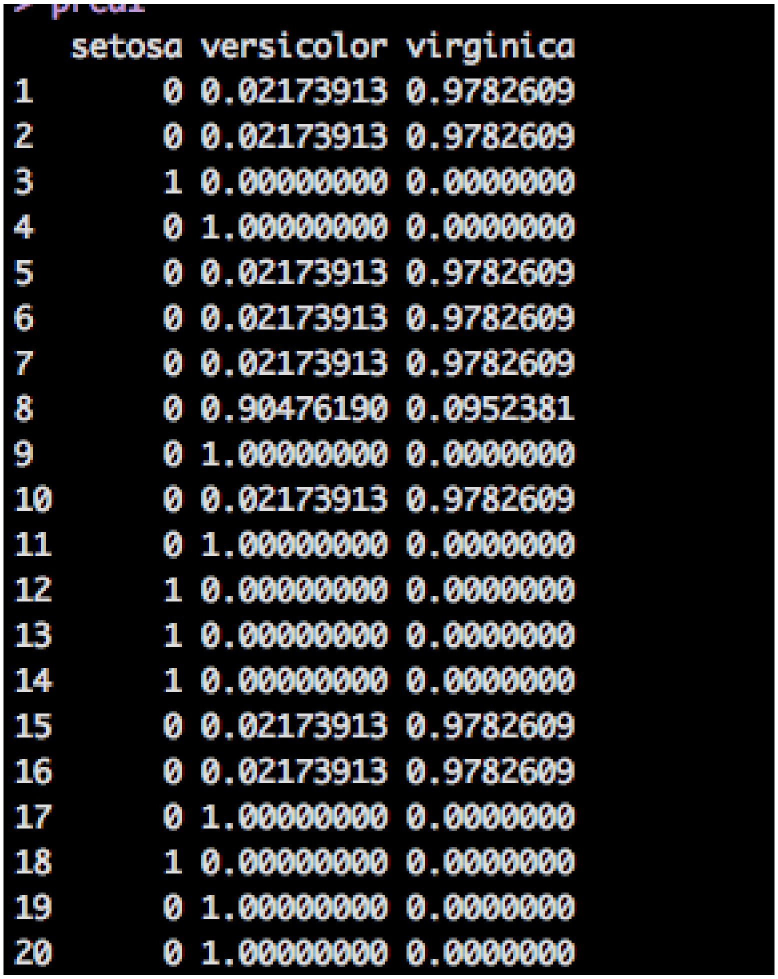
```
> plot(tree1)
> text(tree1)
```



```
> plot(iris$Petal.Width, iris$Sepal.Width, pch = 19, col = as.numeric(iris$Species))
> partition.tree(tree1, label = "Species", add = TRUE)
> legend(1.75, 4.5, legend = unique(iris$Species), col = unique(as.numeric(iris$Species)), pch =
19)
```

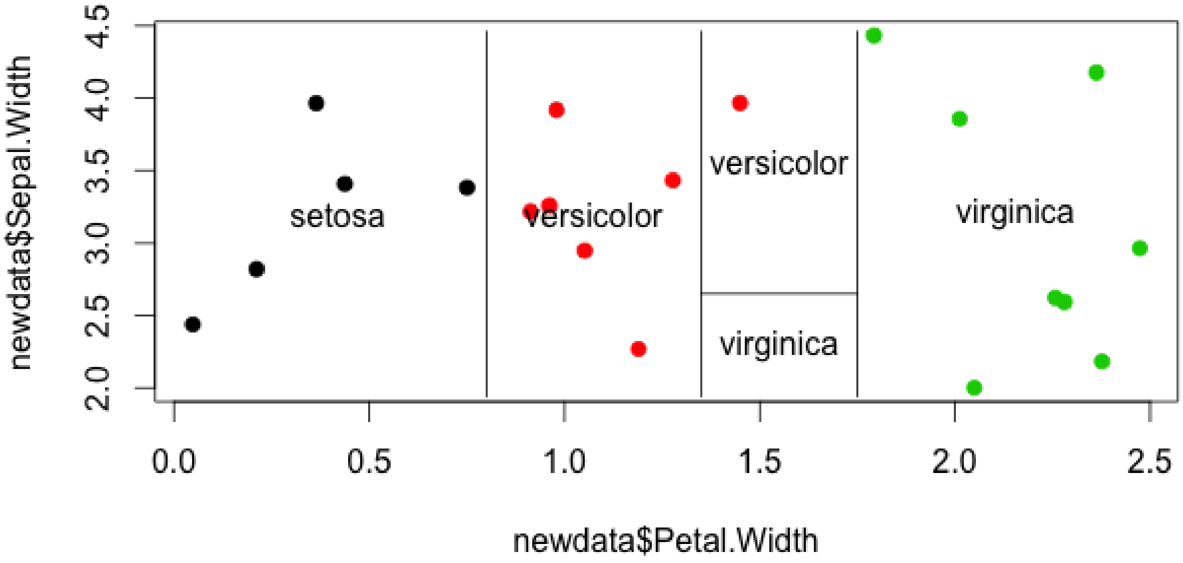


```
> set.seed(32313)

> newdata <- data.frame(Petal.Width = runif(20,0,2.5), Sepal.Width = runif(20,2,4.5))

> pred1 <- predict(tree1, newdata)

> pred1
```



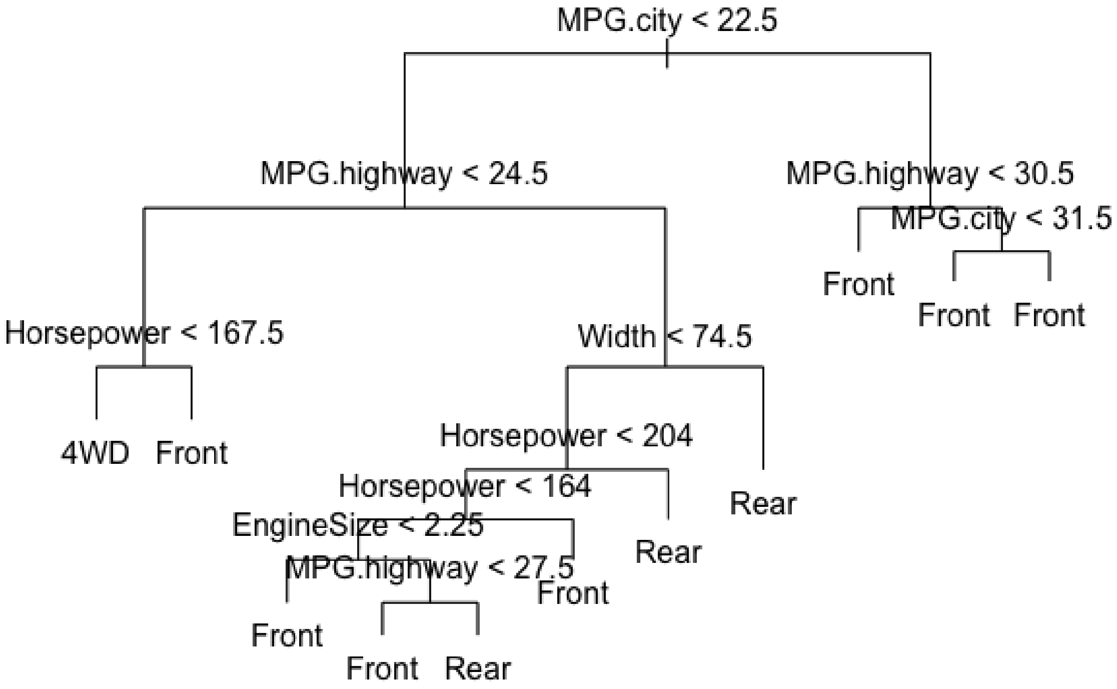
```
> pred1
   setosa versicolor virginica
1       0 0.02173913 0.9782609
2       0 0.02173913 0.9782609
3       1 0.00000000 0.0000000
4       0 1.00000000 0.0000000
5       0 0.02173913 0.9782609
6       0 0.02173913 0.9782609
7       0 0.02173913 0.9782609
8       0 0.90476190 0.0952381
9       0 1.00000000 0.0000000
10      0 0.02173913 0.9782609
11      0 1.00000000 0.0000000
12      1 0.00000000 0.0000000
13      1 0.00000000 0.0000000
14      1 0.00000000 0.0000000
15      0 0.02173913 0.9782609
16      0 0.02173913 0.9782609
17      0 1.00000000 0.0000000
18      1 0.00000000 0.0000000
19      0 1.00000000 0.0000000
20      0 1.00000000 0.0000000
```

```
> pred1 <- predict(tree1, newdata, type = "class")
> plot(newdata$Petal.Width, newdata$Sepal.Width, col = as.numeric(pred1), pch = 19)
> partition.tree(tree1, "Species", add = TRUE)
```

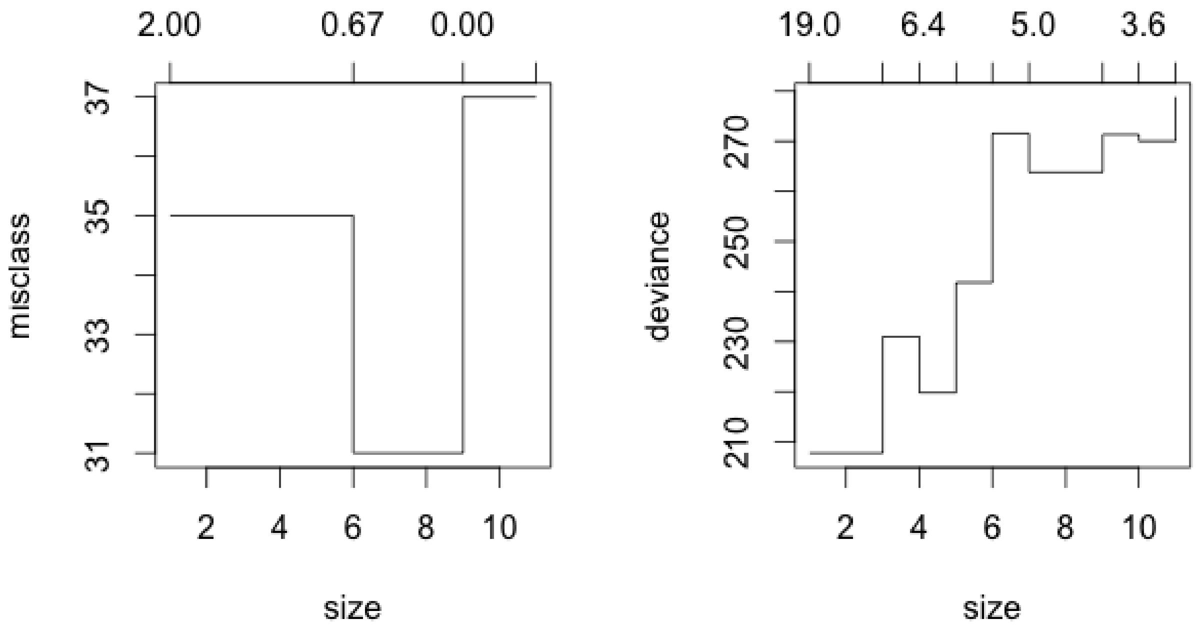


**Part B: Cars93 Data Set**

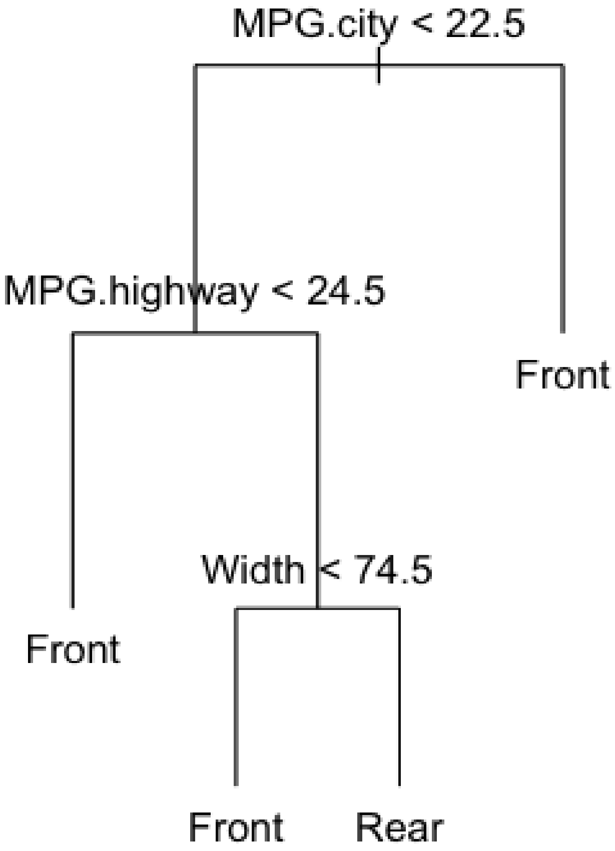
```
> data(Cars93, package = "MASS")
> head(Cars93)
> treeCars <- tree(DriveTrain ~ MPG.city + MPG.highway + AirBags + EngineSize + Width +
Length + Weight + Price + Cylinders + Horsepower + Wheelbase, data=Cars93)
> plot(treeCars)
> text(treeCars)
```



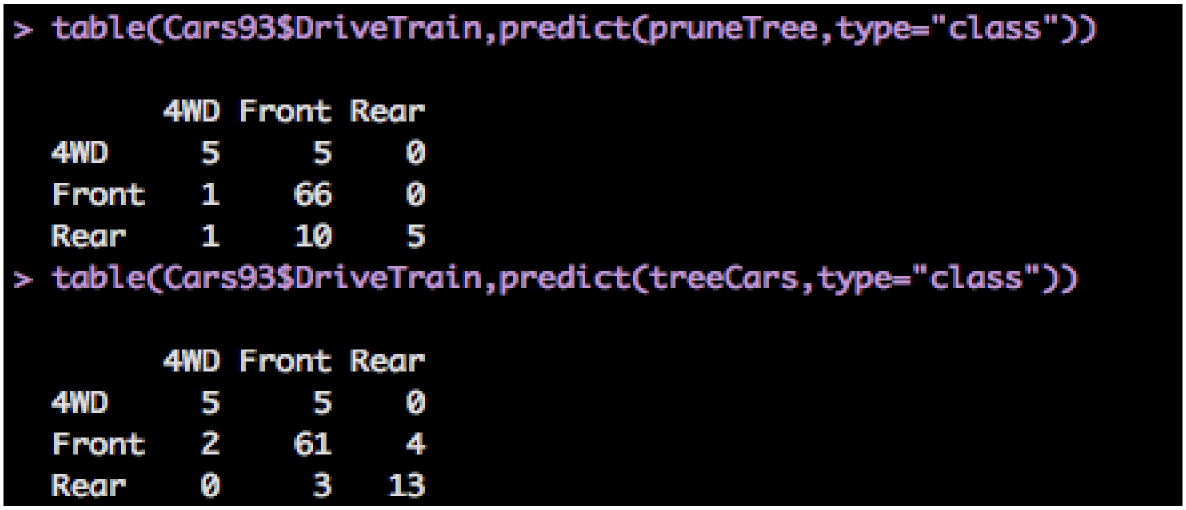
```
> par(mfrow = c(1,2))
> plot(cv.tree(treeCars, FUN = prune.tree, method = "misclass"))
> plot(cv.tree(treeCars))
```



```
> pruneTree <- prune.tree(treeCars, best = 4)
> plot(pruneTree)
> text(pruneTree)
```



```
> table(Cars93$DriveTrain, predict(pruneTree, type = "class"))
> table(Cars93$DriveTrain, predict(treeCars, type = "class"))
```



```
> table(Cars93$DriveTrain,predict(pruneTree,type="class"))
        
         4WD Front Rear
  4WD      5     5    0
  Front    1    66    0
  Rear     1    10    5
> table(Cars93$DriveTrain,predict(treeCars,type="class"))
        
         4WD Front Rear
  4WD      5     5    0
  Front    2    61    4
  Rear     0     3   13
```

## Problem 3: More Trees in R

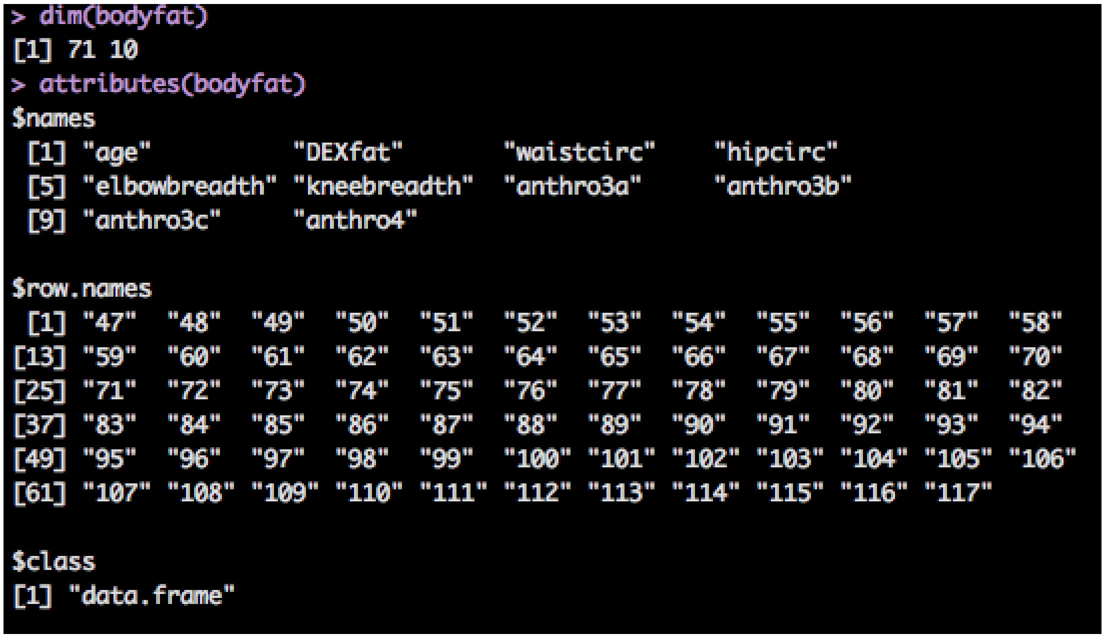
>install.packages("mboost")

>library(mboost)

>data("bodyfat", package = "mboost")

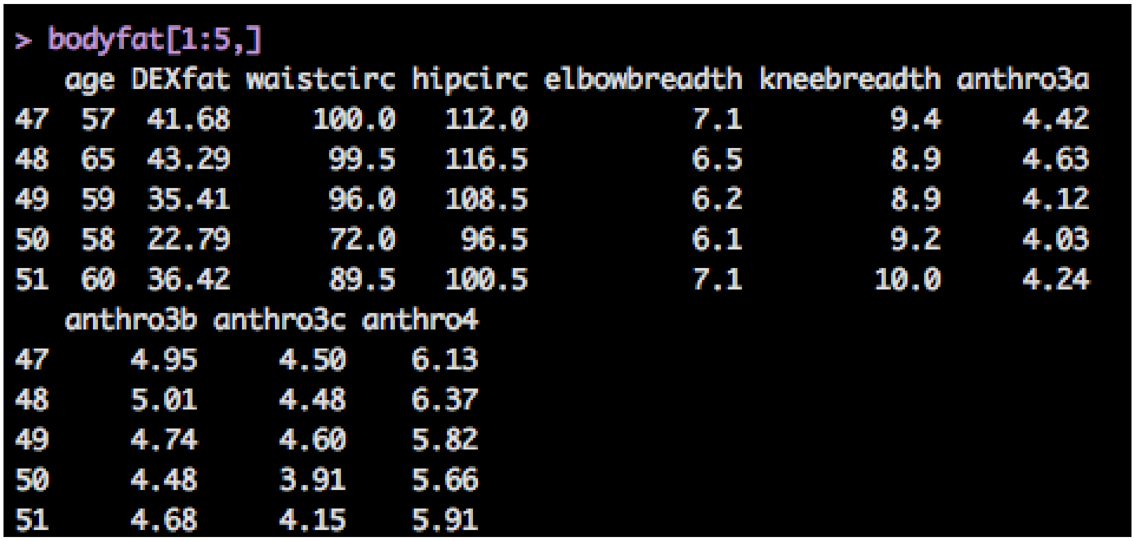
> dim(bodyfat)

> attributes(bodyfat)



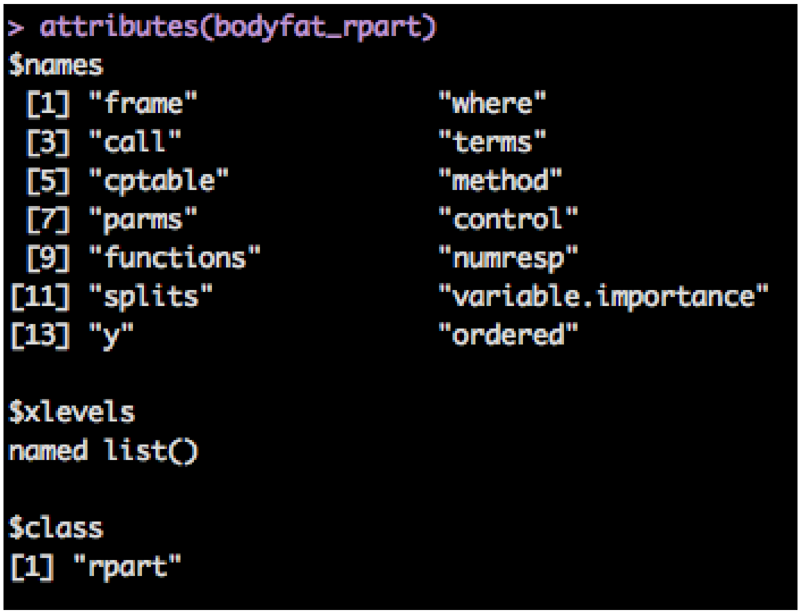
```
> dim(bodyfat)
[1] 71 10
> attributes(bodyfat)
$names
 [1] "age"          "DEXfat"       "waistcirc"    "hipcirc"
 [5] "elbowbreadth" "kneebreadth"  "anthro3a"     "anthro3b"
 [9] "anthro3c"     "anthro4"

$row.names
 [1] "47"  "48"  "49"  "50"  "51"  "52"  "53"  "54"  "55"  "56"  "57"  "58"
[13] "59"  "60"  "61"  "62"  "63"  "64"  "65"  "66"  "67"  "68"  "69"  "70"
[25] "71"  "72"  "73"  "74"  "75"  "76"  "77"  "78"  "79"  "80"  "81"  "82"
[37] "83"  "84"  "85"  "86"  "87"  "88"  "89"  "90"  "91"  "92"  "93"  "94"
[49] "95"  "96"  "97"  "98"  "99"  "100" "101" "102" "103" "104" "105" "106"
[61] "107" "108" "109" "110" "111" "112" "113" "114" "115" "116" "117"

$class
[1] "data.frame"
```

> bodyfat[1:5,]



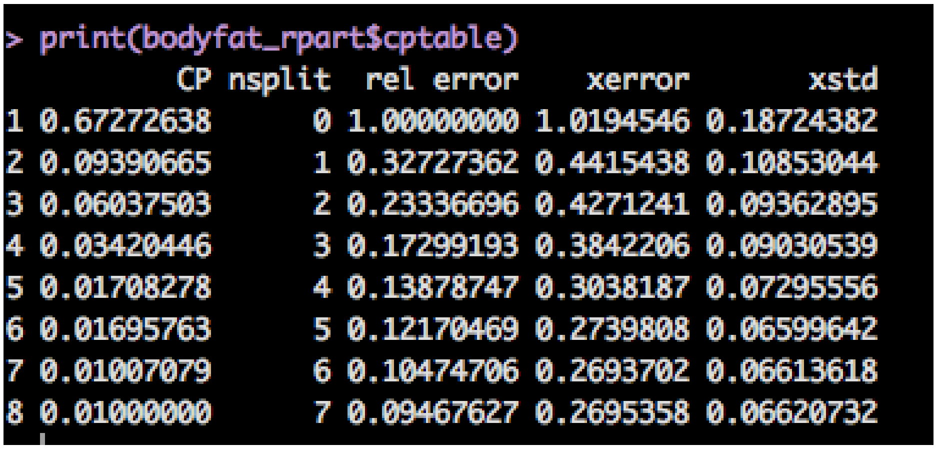
```
> bodyfat[1:5,]
   age DEXfat waistcirc hipcirc elbowbreadth kneebreadth anthro3a
47  57  41.68     100.0   112.0          7.1         9.4     4.42
48  65  43.29      99.5   116.5          6.5         8.9     4.63
49  59  35.41      96.0   108.5          6.2         8.9     4.12
50  58  22.79      72.0    96.5          6.1         9.2     4.03
51  60  36.42      89.5   100.5          7.1        10.0     4.24
   anthro3b anthro3c anthro4
47     4.95     4.50    6.13
48     5.01     4.48    6.37
49     4.74     4.60    5.82
50     4.48     3.91    5.66
51     4.68     4.15    5.91
```

```
> set.seed(1234)
> ind <- sample(2, nrow(bodyfat), replace=TRUE, prob = c(0.7, 0.3))
> bodyfat.train <- bodyfat[ind==1,]
> bodyfat.test <- bodyfat[ind==2,]

> library(rpart)
> myFormula <- DEXfat ~ age + waistcirc + hipcirc + elbowbreadth + kneebreadth
> bodyfat_rpart <- rpart(myFormula, data = bodyfat.train, control = rpart.control(minsplit = 10))
> attributes(bodyfat_rpart)
```



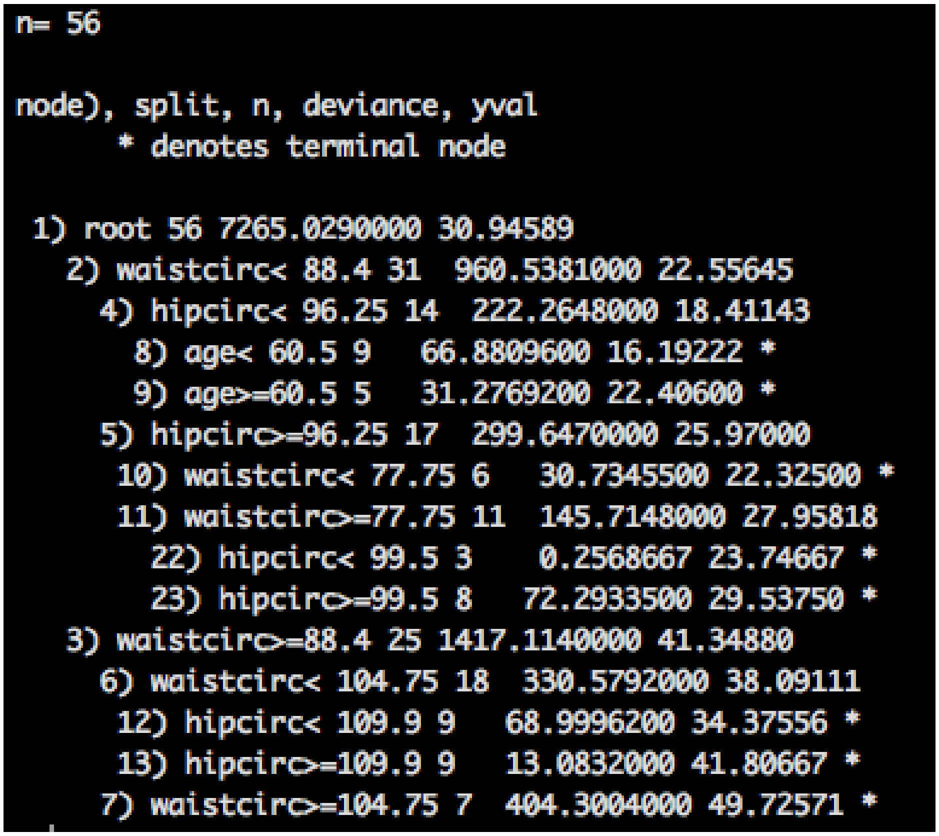
```
> attributes(bodyfat_rpart)
$names
 [1] "frame"                "where"
 [3] "call"                 "terms"
 [5] "cptable"              "method"
 [7] "parms"                "control"
 [9] "functions"            "numresp"
[11] "splits"               "variable.importance"
[13] "y"                    "ordered"

$xlevels
named list()

$class
[1] "rpart"
```

> print(bodyfat_rpart$cptable)



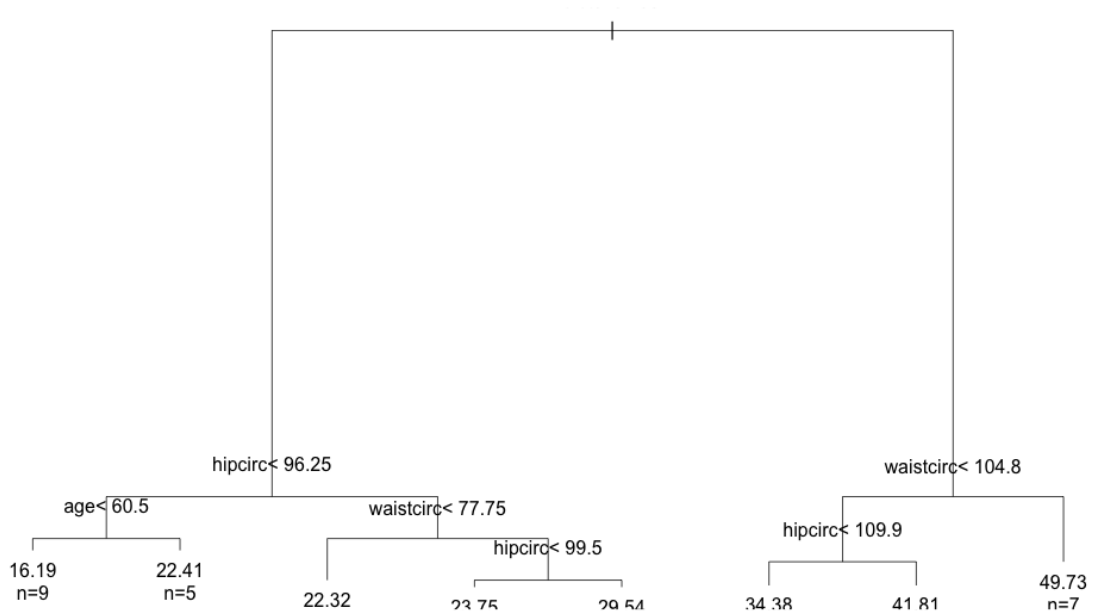
```
> print(bodyfat_rpart$cptable)
          CP nsplit  rel error    xerror       xstd
1 0.67272638      0 1.00000000 1.0194546 0.18724382
2 0.09390665      1 0.32727362 0.4415438 0.10853044
3 0.06037503      2 0.23336696 0.4271241 0.09362895
4 0.03420446      3 0.17299193 0.3842206 0.09030539
5 0.01708278      4 0.13878747 0.3038187 0.07295556
6 0.01695763      5 0.12170469 0.2739808 0.06599642
7 0.01007079      6 0.10474706 0.2693702 0.06613618
8 0.01000000      7 0.09467627 0.2695358 0.06620732
```

> print(bodyfat_rpart)

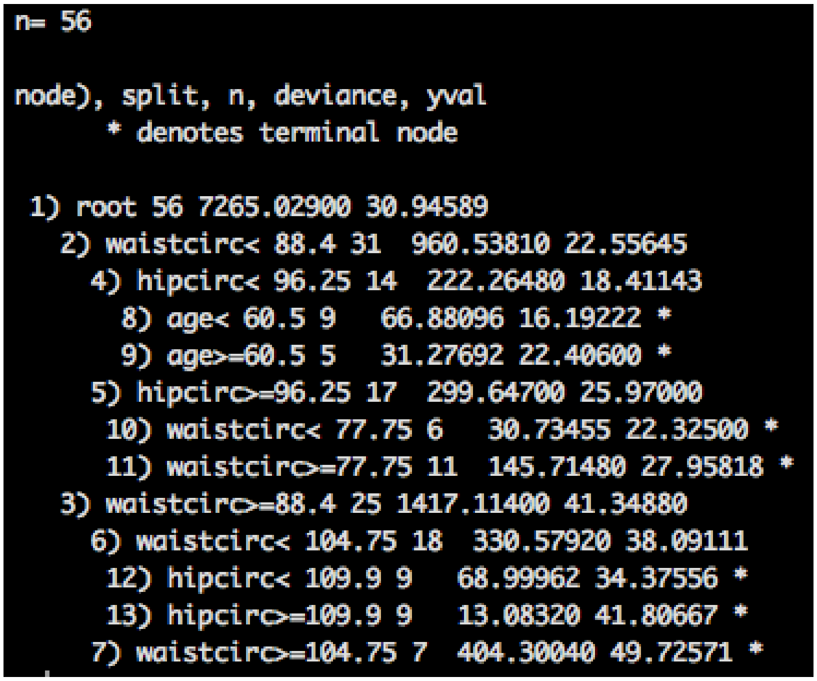


```
n= 56 

node), split, n, deviance, yval
      * denotes terminal node

 1) root 56 7265.0290000 30.94589  
   2) waistcirc< 88.4 31  960.5381000 22.55645  
     4) hipcirc< 96.25 14  222.2648000 18.41143  
       8) age< 60.5 9   66.8809600 16.19222 *
       9) age>=60.5 5   31.2769200 22.40600 *
     5) hipcirc>=96.25 17  299.6470000 25.97000  
      10) waistcirc< 77.75 6   30.7345500 22.32500 *
      11) waistcirc>=77.75 11  145.7148000 27.95818  
        22) hipcirc< 99.5 3    0.2568667 23.74667 *
        23) hipcirc>=99.5 8   72.2933500 29.53750 *
   3) waistcirc>=88.4 25 1417.1140000 41.34880  
     6) waistcirc< 104.75 18  330.5792000 38.09111  
      12) hipcirc< 109.9 9   68.9996200 34.37556 *
      13) hipcirc>=109.9 9   13.0832000 41.80667 *
     7) waistcirc>=104.75 7  404.3004000 49.72571 *
```

```
> plot(bodyfat_rpart)
> text(bodyfat_rpart, use.n=T)
```

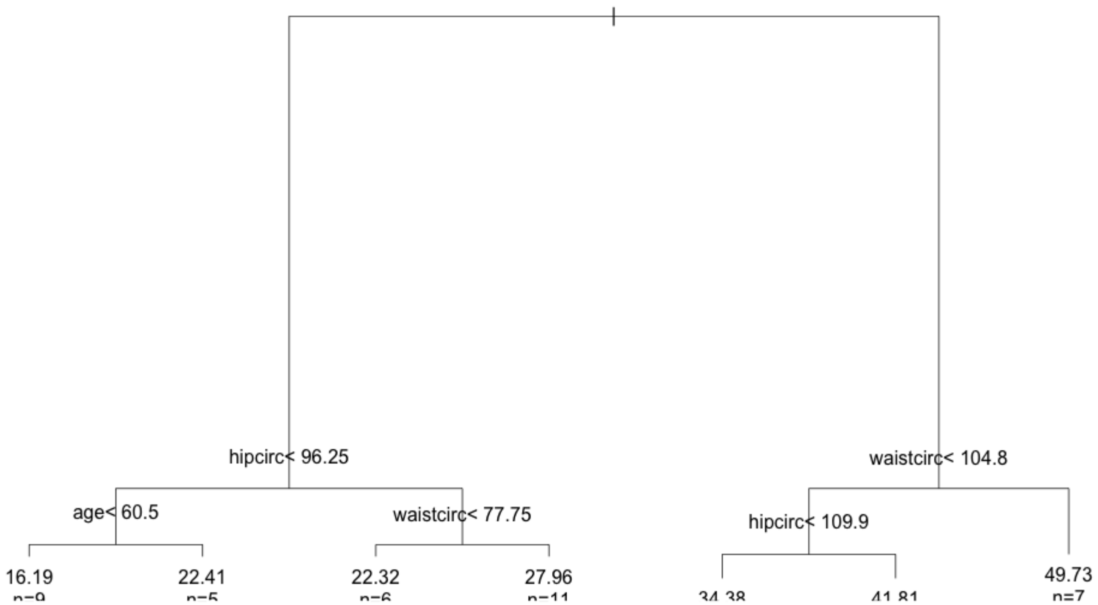


```
> opt <- which.min(bodyfat_rpart$cptable[,"xerror"])
> cp <- bodyfat_rpart$cptable[opt, "CP"]
> bodyfat_prune <- prune(bodyfat_rpart, cp = cp)
> print(bodyfat_prune)
```

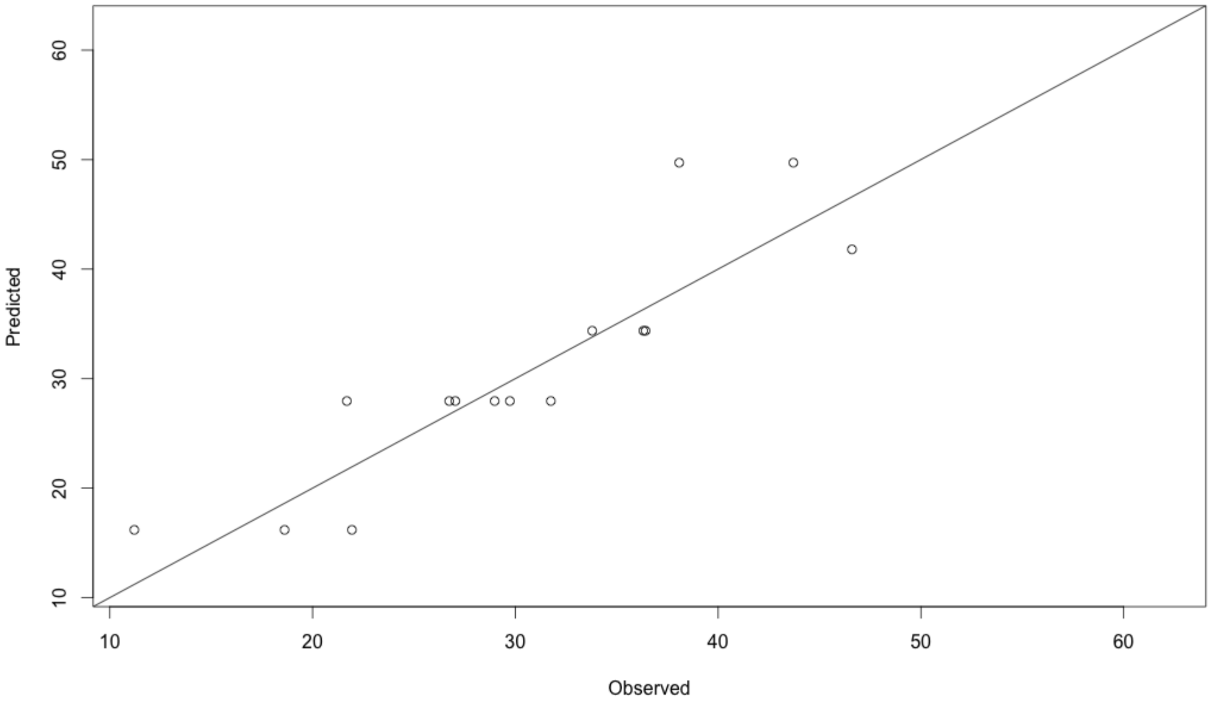


```
n= 56

node), split, n, deviance, yval
      * denotes terminal node

 1) root 56 7265.02900 30.94589
   2) waistcirc< 88.4 31  960.53810 22.55645
     4) hipcirc< 96.25 14  222.26480 18.41143
       8) age< 60.5 9   66.88096 16.19222 *
       9) age>=60.5 5   31.27692 22.40600 *
     5) hipcirc>=96.25 17  299.64700 25.97000
      10) waistcirc< 77.75 6   30.73455 22.32500 *
      11) waistcirc>=77.75 11  145.71480 27.95818 *
   3) waistcirc>=88.4 25 1417.11400 41.34880
     6) waistcirc< 104.75 18  330.57920 38.09111
      12) hipcirc< 109.9 9   68.99962 34.37556 *
      13) hipcirc>=109.9 9   13.08320 41.80667 *
     7) waistcirc>=104.75 7  404.30040 49.72571 *
```

```
> plot(bodyfat_prune)
> text(bodyfat_prune, use.n=T)
```



```
> DEXfat_pred <- predict(bodyfat_prune, newdata=bodyfat.test)
> xlim <- range(bodyfat$DEXfat)
> plot(DEXfat_pred ~ DEXfat, data=bodyfat.test, xlab = "Observed",
+ ylab = "Predicted", ylim = xlim, xlim = xlim)
> abline(a = 0, b = 1)
```


Predicted
Observed