Exercise 2 – Fit a regression tree on the Hitters data

(2 input variables)

```
# sample 70% of the row indices for subsetting as training data
set.seed (1)
trainHit <- sample(1:nrow(Hitters), 0.7*nrow(Hitters))
Hitters.train <- Hitters[trainHit,]
Hitters.test <- Hitters[-trainHit,]

#### Fit a regression tree to predict Salary from *Years and Hits*.
tree.salaryHitters <- tree(Salary ~ Years + Hits, data = Hitters)
# tree() uses binary recursive partitioning.
# The split which maximizes the reduction in impurity is chosen,
# the data set is then split and the process repeated.
# Splitting continues until the terminal nodes are too small or too few to be split
# Tree growth is limited to a depth of 31 by the use of integers to label nodes.

summary(tree.salaryHitters)
    # Output:
    #   - There are 8 terminal nodes (leaves) of the tree.
    #   - Here “residual mean deviance” is just mean squared error (RMS)

# Plot the regression tree
plot(tree.salaryHitters)
text(tree.salaryHitters, cex=0.75) # cex: set character size to 0.75

# Plot the corresponding regions

  # simple plot:
    plot(Hitters$Years,Hitters$Hits,col='steelblue', pch=20, xlab="Years",ylab="Hits")
    partition.tree(tree.salaryHitters,ordvars=c("Years","Hits"),add=TRUE,cex=1)

  # plot with salary value in color code:
    # prepare Salary data for plot
      salary.deciles = quantile(Salary,0:10/10)
      cut.salary = cut(Hitters$Salary,salary.deciles,include.lowest=TRUE)
    # plot the point cloud and regions
      plot(Years,Hits,col=grey(10:2/11)[cut.salary],pch=20,
         xlab="Years",ylab="Hits")
      partition.tree(tree.salaryHitters,ordvars=c("Years","Hits"),add=TRUE,cex=1)

tree.salaryHitters
# node): node number
# split: split criterion, e.g. Thal: normal, or Ca < 0.5
# n: number of observations in that branch
# deviance (the smaller the better)
# yval: overall prediction for the branch (mean value or Yes or No)
# (ybrob): the fraction of observations in that branch that take on values of (Yes No)
# * denotes terminal node
```

See R-code on Moodle!

Exercise 2 – Fit a regression tree on the Hitters data

(All input variables)

```
#### Fit a regression tree to predict Salary from *all other variables*.
tree.salaryHitters <- tree(Salary ~ .-Salary, data = Hitters) # use all variables excep Salary
summary(tree.salaryHitters)

# Plot the regression tree
plot(tree.salaryHitters)
text(tree.salaryHitters, cex=0.75) # cex: set character size to 0.75

# use the tree to make predictions on the *test set*
tree.salaryHitters.pred <- predict(tree.salaryHitters, newdata = Hitters.test)

# compare predictions of regression tree with true values (visually)
  # We plot the predictions against ground truth.
  # A perfect prediction would give a line with intercept 0 and slope 1.
salaryHitters.test <- Hitters.test$Salary
plot(tree.salaryHitters.pred,salaryHitters.test)
abline (0 ,1) # compare with the function f(x)=x (intercept 0, slope 1)

# calculate the mean squared error on *test data*
salaryHitters.test <- Hitters[-trainHit ,"Salary"]
(tree.salaryHitters.MSE <- mean((tree.salaryHitters.pred - salaryHitters.test)^2)) # MSE = 0.146
sqrt(tree.salaryHitters.MSE)
# square root of the MSE = 0.383
# --> test predictions are within around $383 of the true median logarithmized salary.
```