Module 4

Decision Tree using R

```
# loading libraries
library(tree)
library(party)
## Loading required package: grid
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
##
## Loading required package: sandwich
## Loading required package: strucchange
## Loading required package: modeltools
## Loading required package: stats4
# loading data
data(iris)
View(iris)
colnames(iris)
## [1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"
## [5] "Species"
# creating a tree using tree package
myTree <- tree(Species ~ Sepal.Width + Petal.Width, data=iris)
#myTree <- tree(Species ~ Sepal.Width + Petal.Width
#               + Sepal.Length + Petal.Length, data=iris)

plot(myTree)
text(myTree)
```

Prediction using R :-)

```
dim(iris)
## [1] 150   5
dim(iris)[1]
## [1] 150
dim(iris)[2]
## [1] 5
# Create the training Data of 100 records and test Data of 50 records

filteredData <- sample(1:dim(iris)[1],
size=as.integer(dim(iris[1])/3),replace = F)
filteredData
##  [1] 110   5  23  83  15  75 135  84   8  64  54 148  29 129 144  22
9
## [18]  27 141  14  47 104   2  67  76  36 131  48 143  99 127 130 140
121
## [35]  10 138  33  62  96 145  98 136  58 146  49  34  19 113  17  81
filteredData <- sample(1:150, size=50,replace = FALSE)
```

```r
View(iris)
# creating trainData & testData
trainData <- iris[-filteredData,]
View(trainData)
testData <- iris[filteredData,]
View(testData)
library(tree)
myTree <- tree(Species ~ Sepal.Width + Petal.Width, data=trainData) #
model

testData$predictedClass <- predict(myTree, newdata = testData,
type="class")

X <- testData[,c("Sepal.Width", "Petal.Width")]
View(X)
X$myPrediction <- predict(myTree, newdata = X, type="class")
#prediction
View(testData)

table(testData$Species)
##
##     setosa versicolor  virginica
##         15         20         15
table(iris$Species)
##
##     setosa versicolor  virginica
##         50         50         50
table(testData$predictedClass, testData$Species)
##
##             setosa versicolor virginica
##   setosa         15          0         0
##   versicolor      0         19         1
##   virginica       0          1        14
sum(testData$predictedClass == testData$Species)/nrow(testData) # accuracy
## [1] 0.96
library(party)
myTree <- ctree(Species ~ Sepal.Width + Petal.Width, data=trainData) #
model
plot(myTree)
```

```r
testData$predictedClass <- predict(myTree, newdata = testData,
type="response")

X <- testData[,c("Sepal.Width", "Petal.Width")]
View(X)
X$myPrediction <- predict(myTree, newdata = X, type="class")
## Error: 'arg' should be one of "response", "node", "prob"
#prediction
View(testData)

table(testData$Species)
##
##     setosa versicolor  virginica
##         15         20         15
table(iris$Species)
##
##     setosa versicolor  virginica
```

```
##            50          50          50
table(testData$predictedClass, testData$Species)
##
##              setosa versicolor virginica
##   setosa        15          0         0
##   versicolor     0         20         1
##   virginica      0          0        14
sum(testData$predictedClass == testData$Species)/nrow(testData) # accuracy
## [1] 0.98
plot(testData$Sepal.Width, testData$Petal.Width,
col=testData$predictedClass)
```