**Assignment 3**

**-Sai Sree Mithra Sripathi**

* **Load “lenses.csv” dataset and store it into a local R variable named “lenses” using “read.csv” R command.**

Took the dataset from mentioned source. Converted to csv and the csv has 24 entries , first entry is the serial number followed by the attributes. The details from lenses.names explains the attributes in the data:

Eg: 1 1 1 1 1 3

1 is serial number

The next four are age – young(1), spectacle prescription(1)-myope, astigmatic(1)-no, tear production rate(1) – reduced and finally 3 is for 3 : the patient should not be fitted with contact lenses.

-- 3 Classes

1 : the patient should be fitted with hard contact lenses,

2 : the patient should be fitted with soft contact lenses,

3 : the patient should not be fitted with contact lenses.

1. age of the patient: (1) young, (2) pre-presbyopic, (3) presbyopic

2. spectacle prescription: (1) myope, (2) hypermetrope

3. astigmatic: (1) no, (2) yes

4. tear production rate: (1) reduced, (2) normal

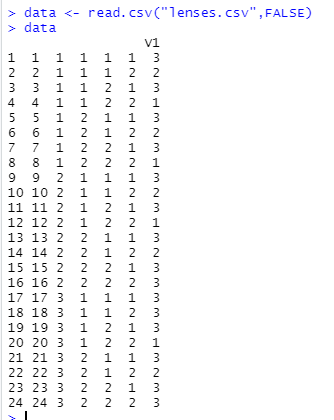
This is the information about the data, let us load the data into a variable and run the basic methods on it.

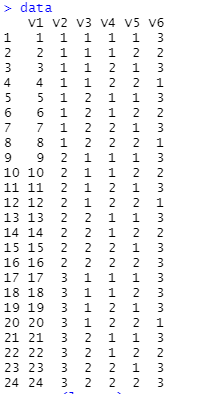
* **Display the content of the “lenses” variable by typing the variable name in RStudio. Note that all data values in the “lenses” variable are numeric.**

Commands:

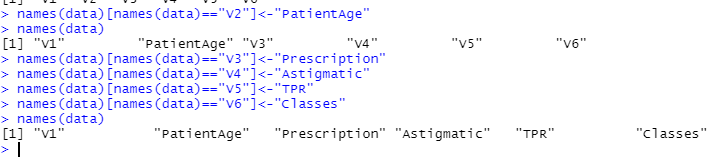
data <- read.csv("lenses.csv",FALSE)

data





* **Clean up the data in “lenses” by replacing all numeric values with descriptive labels as outlined in the “lenses data description” file linked to Week 3 module. In this data cleaning activity, you can use the “replace” R command.**



Commands:

names(data)

names(data)[names(data)=="V2"]<-"PatientAge"

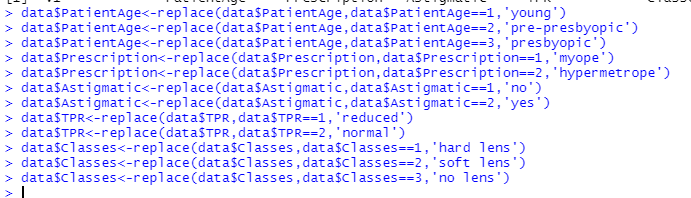
names(data)[names(data)=="V3"]<-"Prescription"

names(data)[names(data)=="V4"]<-"Astigmatic"

names(data)[names(data)=="V5"]<-"TPR"

names(data)[names(data)=="V6"]<-"Classes"

names(data)



Commands to replace numeric data with values:

data$PatientAge<-replace(data$PatientAge,data$PatientAge==1,'young')

data$PatientAge<-replace(data$PatientAge,data$PatientAge==2,'pre-presbyopic')

data$PatientAge<-replace(data$PatientAge,data$PatientAge==3,'presbyopic')

data$Prescription<-replace(data$Prescription,data$Prescription==1,'myope')

data$Prescription<-replace(data$Prescription,data$Prescription==2,'hypermetrope')

data$Astigmatic<-replace(data$Astigmatic,data$Astigmatic==1,'no')

data$Astigmatic<-replace(data$Astigmatic,data$Astigmatic==2,'yes')

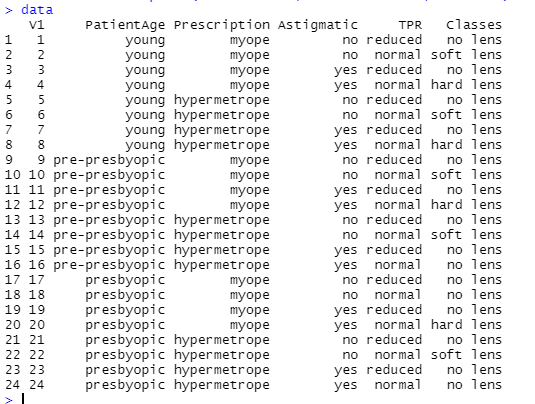
data$TPR<-replace(data$TPR,data$TPR==1,'reduced')

data$TPR<-replace(data$TPR,data$TPR==2,'normal')

data$Classes<-replace(data$Classes,data$Classes==1,'hard lens')

data$Classes<-replace(data$Classes,data$Classes==2,'soft lens')

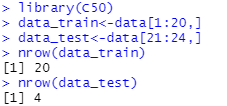
data$Classes<-replace(data$Classes,data$Classes==3,'no lens')



* **Build the decision tree model using the following steps:**
  1. **Create two sets of data; training data and testing data. For this activity, use the seed value of “10203”, and use “20” instances/rows out of the “24” instances/rows for training data and the remaining “4” instances/rows for testing data.**
  2. **Train the decision tree. Evaluate the decision tree model.**

install.packages("C50")

library(C50)



set.seed(10203)

str(data)

str(data\_train)

str(data\_test)

data\_train$Classes<-as.factor(data\_train$Classes)

data\_train$PatientAge<-as.factor(data\_train$PatientAge)

data\_train$Prescription<-as.factor(data\_train$Prescription)

data\_train$Astigmatic<-as.factor(data\_train$Astigmatic)

data\_train$TPR<-as.factor(data\_train$TPR)

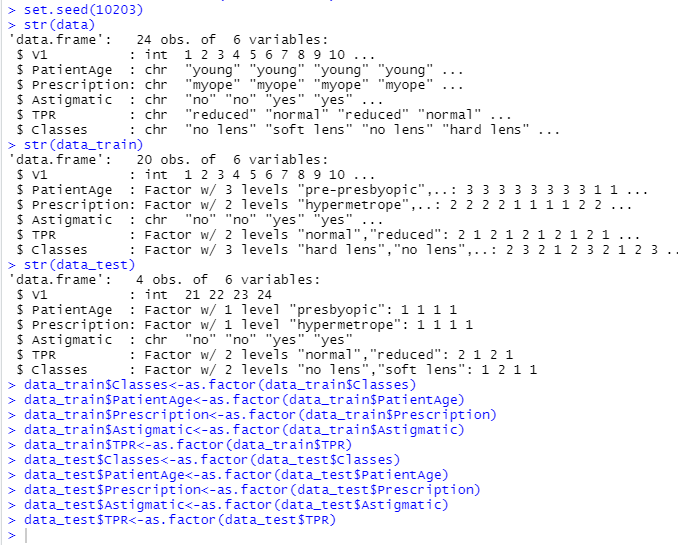
data\_test$Classes<-as.factor(data\_test$Classes)

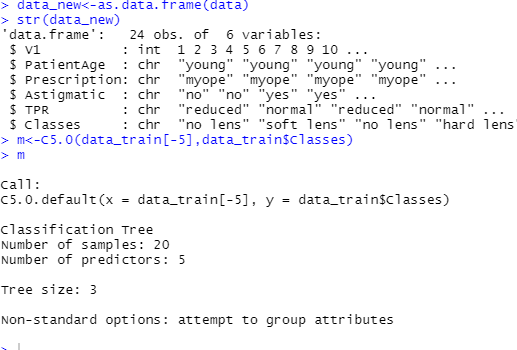
data\_test$PatientAge<-as.factor(data\_test$PatientAge)

data\_test$Prescription<-as.factor(data\_test$Prescription)

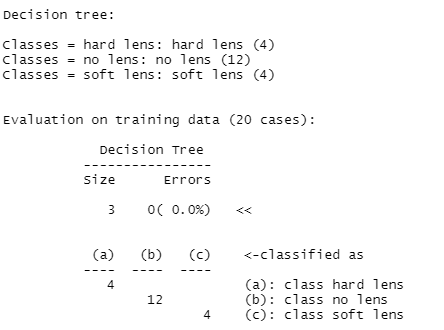
data\_test$Astigmatic<-as.factor(data\_test$Astigmatic)

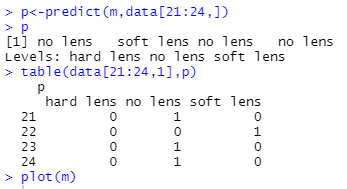
data\_test$TPR<-as.factor(data\_test$TPR)

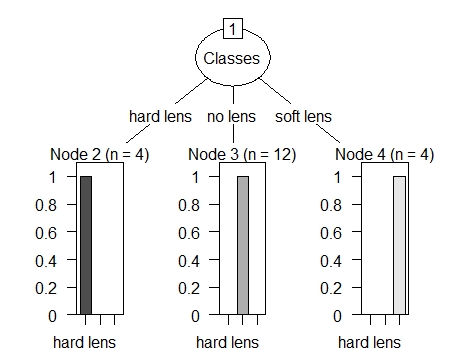




Summary(m)







From the training data of 20 cases – 12 are no lens , 4 hard lens and 4 soft lens, meaning there is almost 75% chances for no lens. When we run the predict on test data of size 4, 3 of them are no lens and one soft lens. Based on test results the analysis was almost correct and accurate. Accuracy of a model can be calculated using ‘rpart’ for the decision tree.

The main challenge for me was during figuring out the columns and assigning column names. I also needed more study on analysing the training data and generating a plot.