Drugs Decision Trees Project

Kiro Shenouda

3/26/2022

My goal for this project is to create a decision tree that helps patients decide which specific drug to take based on different characteristics like age, sex, blood pressure, cholesterol level, as well as sodium and potassium levels.

Load the Libraries

```
library(tree)
library(rpart)
library(rpart.plot)
library(caTools)
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

Get the Data

```
drugs = read.csv('/Users/kiroshenouda/Desktop/COMPSCI/R ML DATASETS/drug200.csv')
head(drugs)
```

```
##   Age Sex     BP Cholesterol Na_to_K  Drug
## 1  23   F   HIGH        HIGH  25.355 drugY
## 2  47   M    LOW        HIGH  13.093 drugC
## 3  47   M    LOW        HIGH  10.114 drugC
## 4  28   F NORMAL        HIGH   7.798 drugX
## 5  61   F    LOW        HIGH  18.043 drugY
## 6  22   F NORMAL        HIGH   8.607 drugX
```

```
str(drugs)
```

```
## 'data.frame':    200 obs. of  6 variables:
##  $ Age        : int  23 47 47 28 61 22 49 41 60 43 ...
##  $ Sex        : chr  "F" "M" "M" "F" ...
##  $ BP         : chr  "HIGH" "LOW" "LOW" "NORMAL" ...
##  $ Cholesterol: chr  "HIGH" "HIGH" "HIGH" "HIGH" ...
##  $ Na_to_K    : num  25.4 13.1 10.1 7.8 18 ...
##  $ Drug       : chr  "drugY" "drugC" "drugC" "drugX" ...
```

```
summary(drugs)
```

```
##       Age            Sex                 BP            Cholesterol
##  Min.   :15.00   Length:200         Length:200         Length:200
##  1st Qu.:31.00   Class :character   Class :character   Class :character
##  Median :45.00   Mode  :character   Mode  :character   Mode  :character
##  Mean   :44.31
##  3rd Qu.:58.00
##  Max.   :74.00
##     Na_to_K           Drug
##  Min.   : 6.269   Length:200
##  1st Qu.:10.445   Class :character
##  Median :13.937   Mode  :character
##  Mean   :16.084
##  3rd Qu.:19.380
##  Max.   :38.247
```

Data Cleaning

```
colSums(is.na(drugs))
```

```
##         Age         Sex          BP Cholesterol     Na_to_K        Drug
##           0           0           0           0           0           0
```

```
sum(is.na(drugs))
```

```
## [1] 0
```

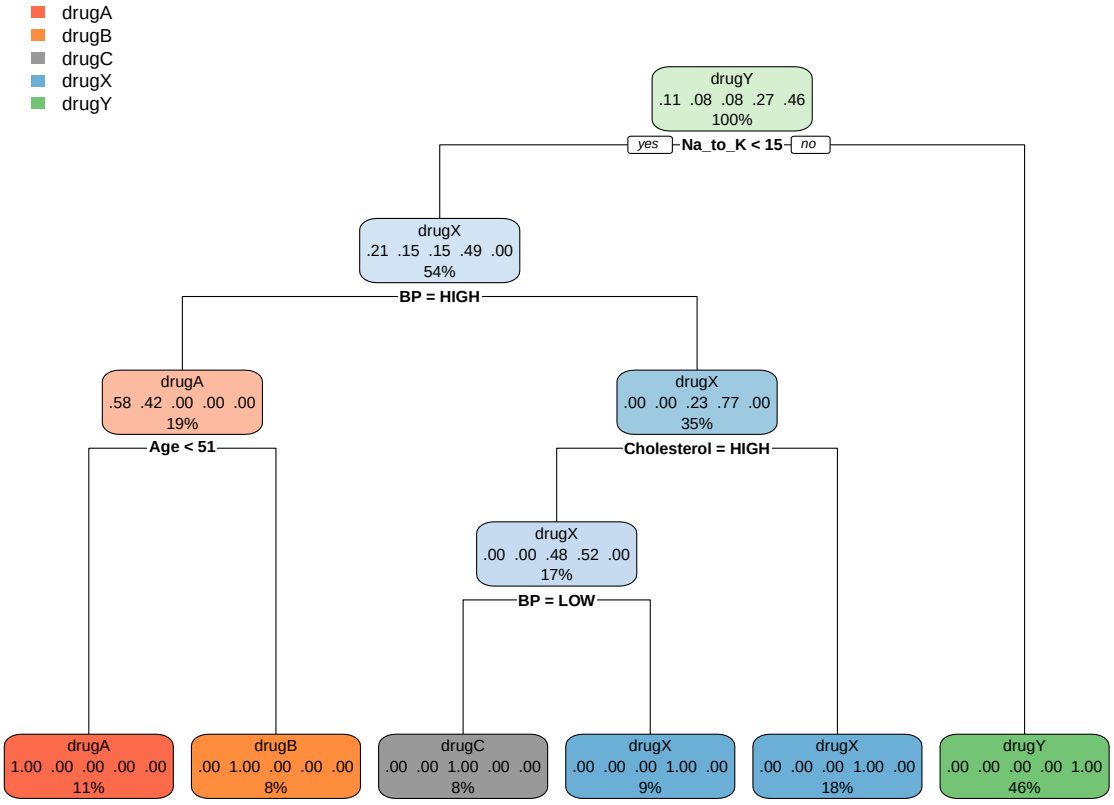
```
drugs$Sex = factor(drugs$Sex)
drugs$BP = factor(drugs$BP)
drugs$Cholesterol = factor(drugs$Cholesterol)
drugs$Drug = factor(drugs$Drug)
str(drugs)
```

```
## 'data.frame':    200 obs. of  6 variables:
##  $ Age        : int  23 47 47 28 61 22 49 41 60 43 ...
##  $ Sex        : Factor w/ 2 levels "F","M": 1 2 2 1 1 1 1 2 2 2 ...
##  $ BP         : Factor w/ 3 levels "HIGH","LOW","NORMAL": 1 2 2 3 2 3 3 2 3 2 ...
##  $ Cholesterol: Factor w/ 2 levels "HIGH","NORMAL": 1 1 1 1 1 1 1 1 1 2 ...
##  $ Na_to_K    : num  25.4 13.1 10.1 7.8 18 ...
##  $ Drug       : Factor w/ 5 levels "drugA","drugB",..: 5 3 3 4 5 4 5 3 5 5 ...
```

Creating the First Tree

```
sample = sample.split(drugs$Drug, SplitRatio =  0.8)
train = subset(drugs, sample == TRUE)
test = subset(drugs, sample == FALSE)
```

```
tree1 = rpart(Drug ~., data = train, method = 'class')
rpart.plot(tree1)
```



Confusion Matrix for Training and Testing Data

```
tree1.preds = predict(tree1, train[1:5], type = 'class')
confusionMatrix(tree1.preds, train$Drug)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction drugA drugB drugC drugX drugY
##      drugA    18     0     0     0     0
##      drugB     0    13     0     0     0
##      drugC     0     0    13     0     0
##      drugX     0     0     0    43     0
##      drugY     0     0     0     0    73
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.9772, 1)
##     No Information Rate : 0.4562
##     P-Value [Acc > NIR] : < 2.2e-16
```

```
##
##                  Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: drugA Class: drugB Class: drugC Class: drugX
## Sensitivity                1.0000      1.00000      1.00000       1.0000
## Specificity                1.0000      1.00000      1.00000       1.0000
## Pos Pred Value             1.0000      1.00000      1.00000       1.0000
## Neg Pred Value             1.0000      1.00000      1.00000       1.0000
## Prevalence                 0.1125      0.08125      0.08125       0.2687
## Detection Rate             0.1125      0.08125      0.08125       0.2687
## Detection Prevalence       0.1125      0.08125      0.08125       0.2687
## Balanced Accuracy          1.0000      1.00000      1.00000       1.0000
##                      Class: drugY
## Sensitivity                1.0000
## Specificity                1.0000
## Pos Pred Value             1.0000
## Neg Pred Value             1.0000
## Prevalence                 0.4562
## Detection Rate             0.4562
## Detection Prevalence       0.4562
## Balanced Accuracy          1.0000
```

```
tree1.preds2 = predict(tree1, test[1:5], type = 'class')
confusionMatrix(tree1.preds2, test$Drug)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction drugA drugB drugC drugX drugY
##      drugA     5     0     0     0     0
##      drugB     0     3     0     0     0
##      drugC     0     0     3     0     0
##      drugX     0     0     0    11     0
##      drugY     0     0     0     0    18
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.9119, 1)
##     No Information Rate : 0.45
##     P-Value [Acc > NIR] : 1.344e-14
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: drugA Class: drugB Class: drugC Class: drugX
## Sensitivity                 1.000        1.000        1.000        1.000
```

```
## Specificity                 1.000        1.000        1.000        1.000
## Pos Pred Value              1.000        1.000        1.000        1.000
## Neg Pred Value              1.000        1.000        1.000        1.000
## Prevalence                  0.125        0.075        0.075        0.275
## Detection Rate              0.125        0.075        0.075        0.275
## Detection Prevalence        0.125        0.075        0.075        0.275
## Balanced Accuracy           1.000        1.000        1.000        1.000
##                       Class: drugY
## Sensitivity                  1.00
## Specificity                  1.00
## Pos Pred Value               1.00
## Neg Pred Value               1.00
## Prevalence                   0.45
## Detection Rate               0.45
## Detection Prevalence         0.45
## Balanced Accuracy            1.00
```

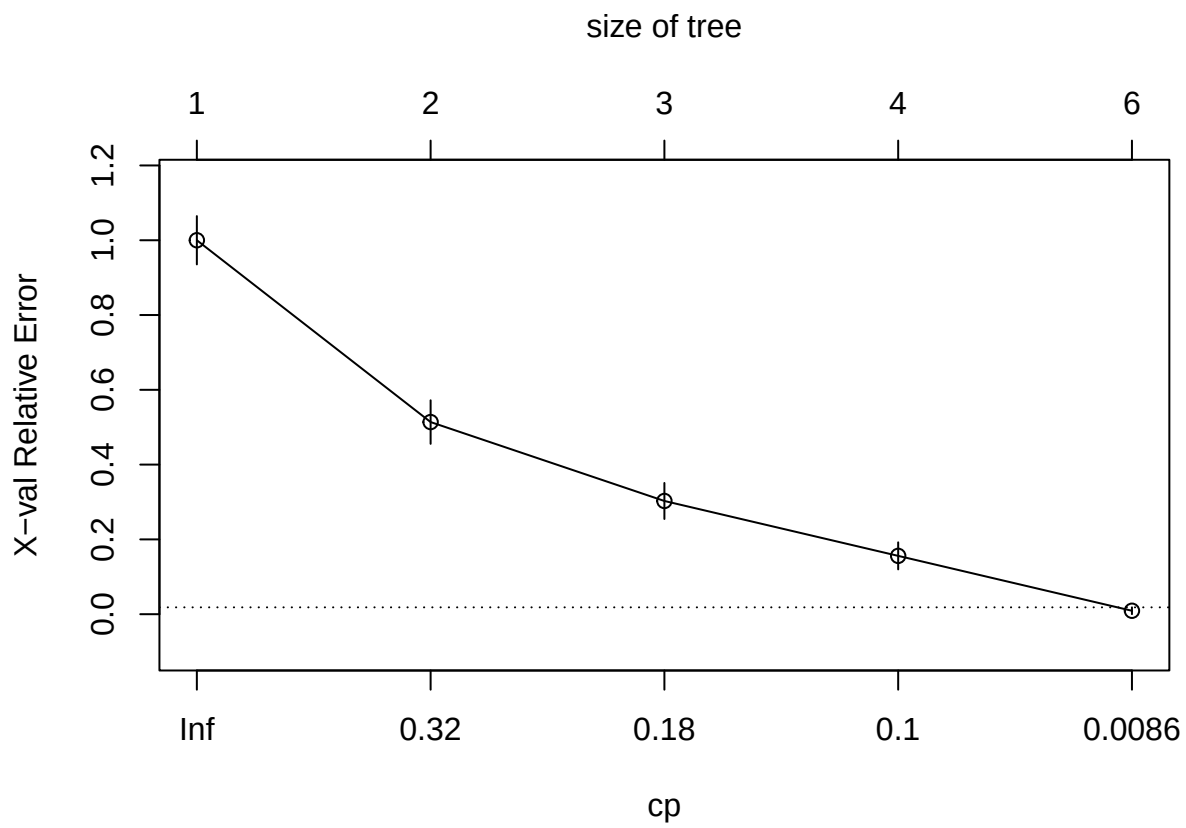
Both the training and testing model have 100% accuracy, as all drugs were predicted correctly.

Creating a Pruned Tree

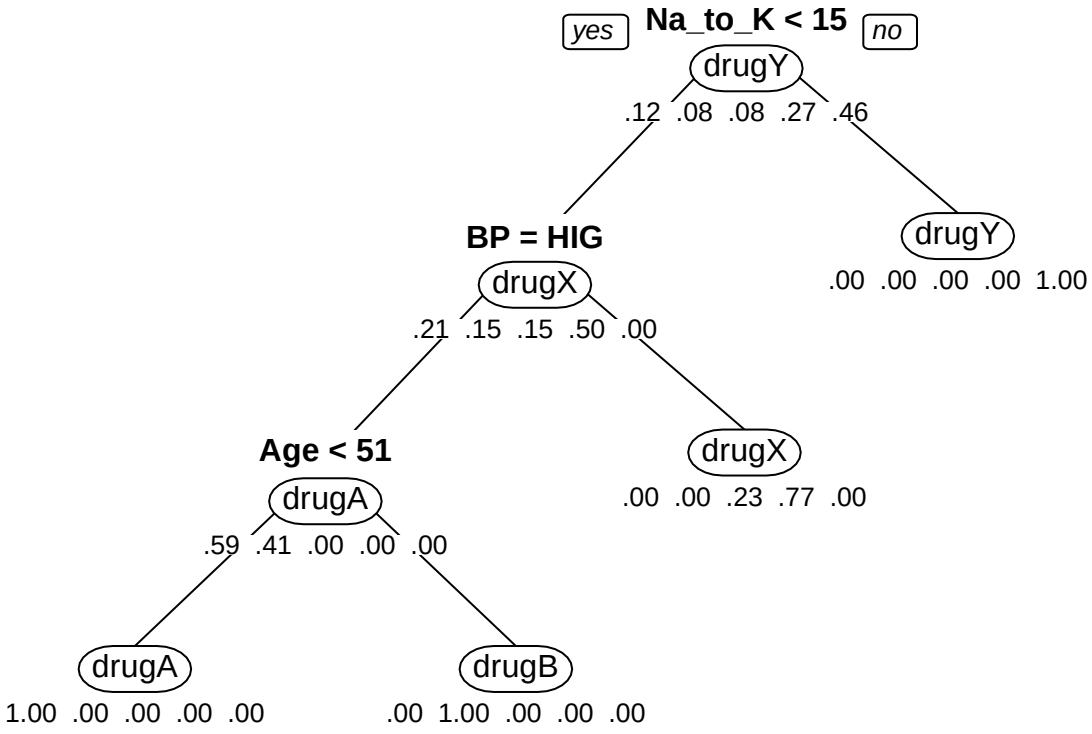
```
tree2 = rpart(Drug ~., data = drugs, method = 'class',
              control = rpart.control(cp = 0.001, minsplit = 2))
tree2$cptable
```

```
##          CP nsplit rel error      xerror        xstd
## 1 0.4954128      0 1.0000000 1.000000000 0.064608915
## 2 0.2110092      1 0.5045872 0.513761468 0.058255075
## 3 0.1467890      2 0.2935780 0.302752294 0.048158584
## 4 0.0733945      3 0.1467890 0.155963303 0.036183328
## 5 0.0010000      5 0.0000000 0.009174312 0.009151347
```

```
plotcp(tree2)
```




```
pruned_tree = prune(tree2, cp = 0.1)
prp(pruned_tree, type = 1, extra = 4, under = TRUE)
```



The root node is the sodium and potassium levels. This indicates that sodium and potassium levels are the most important predictor for deciding which drug to take. The next two decision nodes are blood pressure and age. If a patient's sodium and potassium level is over 15, they should take drug Y. If sodium and potassium level is less than 15 and if they have low blood pressure, they should take drug X. If they have high blood pressure and are older than 51 years, they should take drug B. If not, they should take drug A.

Sources

[1] https://www.kaggle.com/datasets/pablomgomez21/drugs-a-b-c-x-y-for-decision-trees?select=drug200.csv