CS771A Assignment 1: Decision Trees

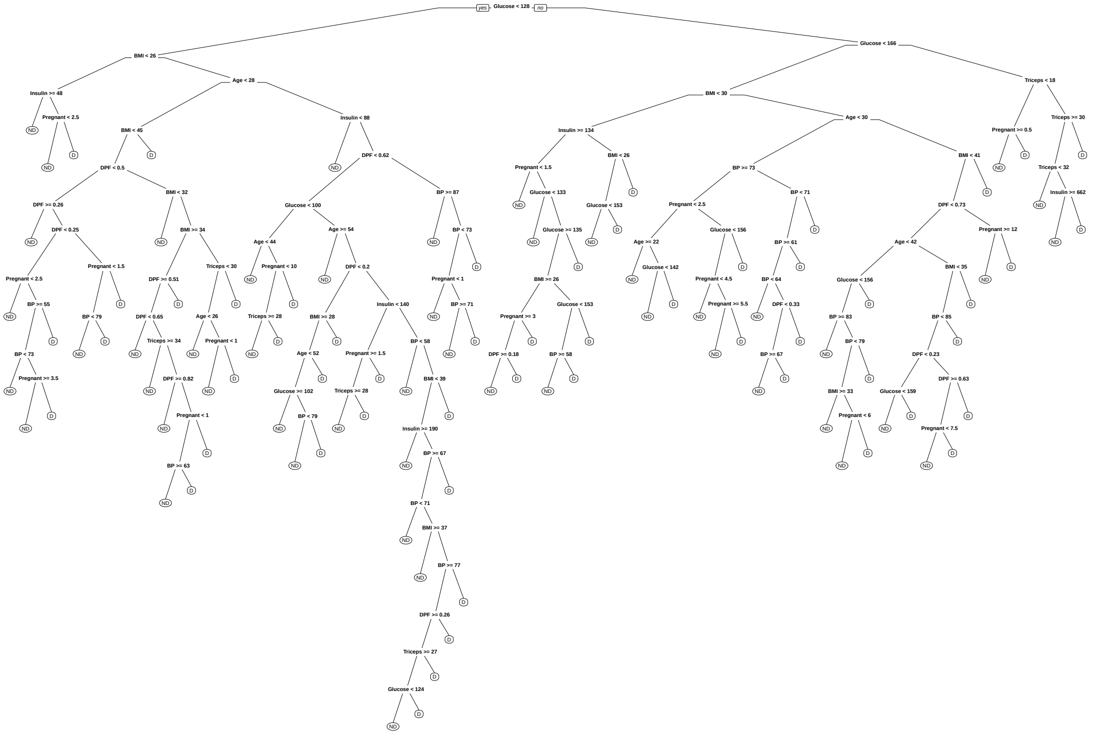
Saurav Kumar (12641)

January 19, 2014

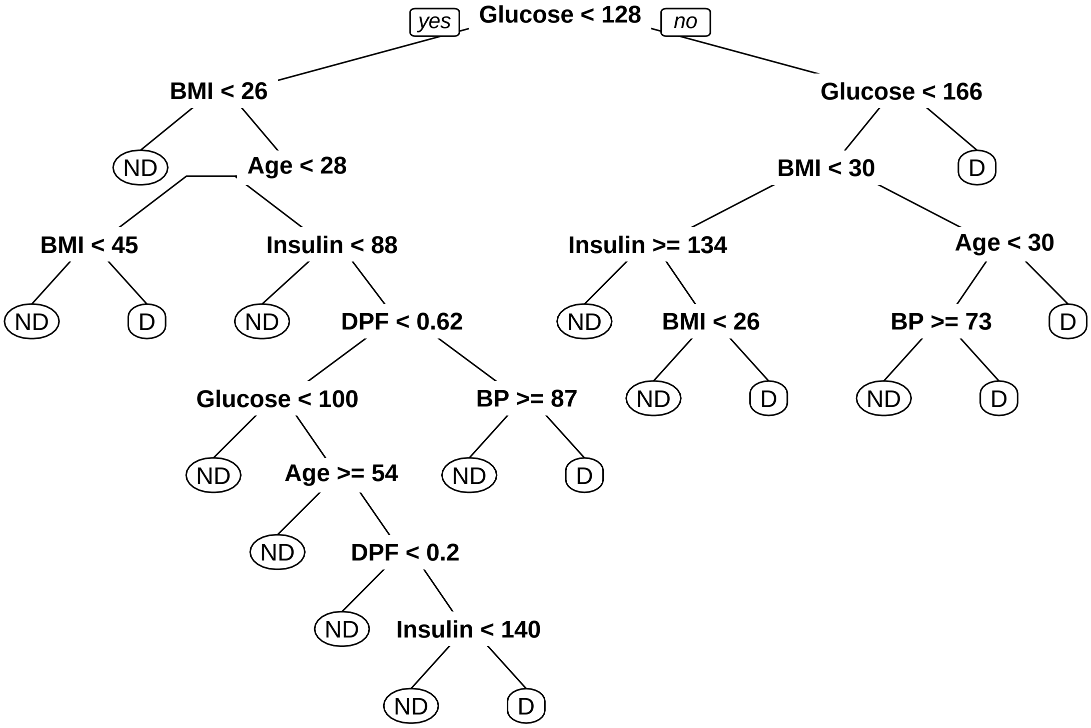
```r
library(rpart)
library(rpart.plot)
set.seed(10)
rawData = read.csv(file="data", header=F, sep=",")
originalData = rawData[sample(nrow(rawData)),]
colnames(originalData) = c("PregnantCount","Glucose","BP","Triceps",
                            "Insulin","BMI","DPF","Age","Class")
N = nrow(originalData)
K = 5
foldWidth = floor(N/K)
Accuracy = 0
for (i in (1:K))
{
    data = originalData
    start = as.integer((i-1)*foldWidth)+1
    end = as.integer(i*foldWidth)
    if(i==K)
    {
        end = N
    }
    testData = data[c(start:end),]
    learnData = data[c(-start:-end),]

    nonZerosCount = colSums(learnData!=0)
    meanVals = colSums(learnData)/nonZerosCount
    learnData$Glucose[learnData$Glucose==0] = meanVals["Glucose"]
    learnData$BP[learnData$BP==0] = meanVals["BP"]
    learnData$Triceps[learnData$Triceps==0] = meanVals["Triceps"]
    learnData$Insulin[learnData$Insulin==0] = meanVals["Insulin"]
    learnData$BMI[learnData$BMI==0] = meanVals["BMI"]

    testData$Glucose[testData$Glucose==0] = NA
    testData$BP[testData$BP==0] = NA
    testData$Triceps[testData$Triceps==0] = NA
    testData$Insulin[testData$Insulin==0] = NA
    testData$BMI[testData$BMI==0] = NA
    diabStat = factor(learnData$Class, levels=0:1, labels=c('ND','D'))
    cfit = rpart(
                diabStat ~ PregnantCount+Glucose+BP+Triceps+Insulin+BMI+DPF+Age,
                data = learnData,
                na.action = na.rpart,
                method ='class',
                parms = list(split = "information"),
                control = rpart.control(
                                            # Grow max possible tree
                                            cp = 0.0,
                                            minsplit = 1,    # Min no. of obs. for which the routine
```

```r
                                              minbucket = 1,  # Min no. of obs in leaf. Default = min
                                          )
                )
    # Pruning
    opt = cfit$cptable[which.min(cfit$cptable[,"xerror"]),"CP"]
    prunedTree = prune(cfit, cp = opt)

    predictedFactor = predict(prunedTree, testData, type="class")
    predictedFrame = as.data.frame.factor(predictedFactor)
    predicted = c(predictedFrame[ ,1]) - 1
    actual = testData$Class
    TP = sum(predicted & actual)
    TN = nrow(testData) - sum(predicted | actual)
    # Accuracy
    print((TP+TN)/nrow(testData))
    Accuracy = Accuracy + (TP+TN)/nrow(testData)
    print("Unpruned Tree")
    rpart.plot(cfit)
    print("Pruned Tree")
    rpart.plot(prunedTree)
}
```

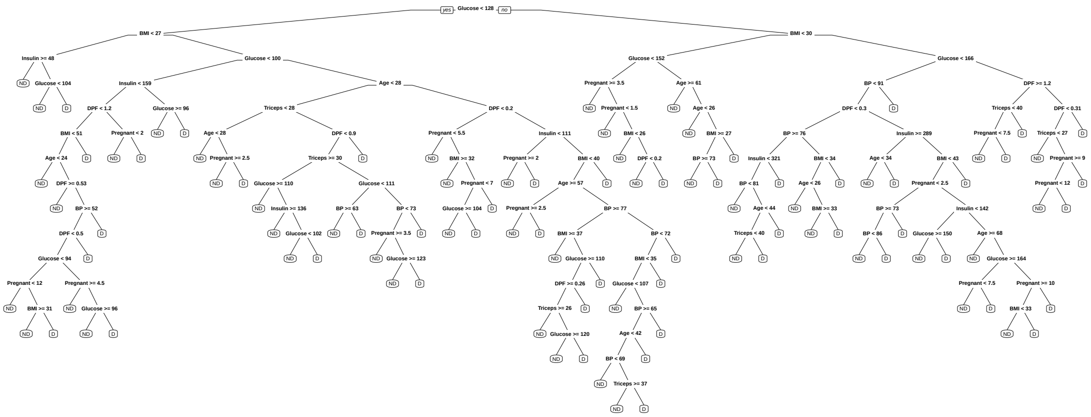
```
## [1] 0.7581699
## [1] "Unpruned Tree"
```



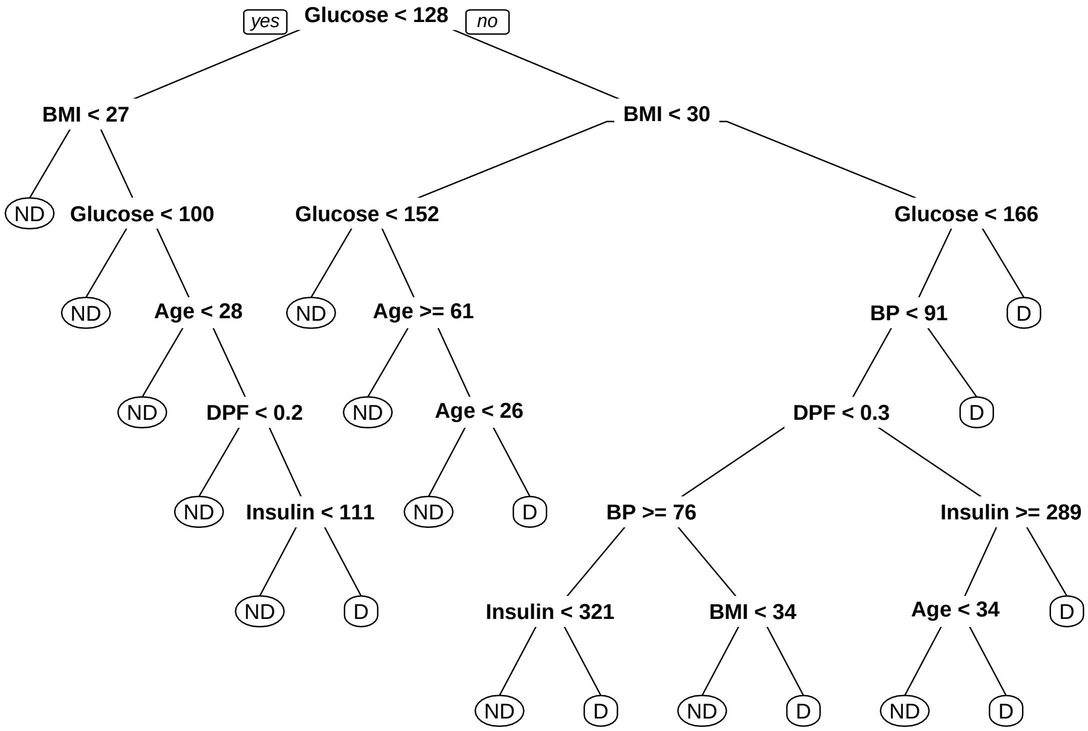
```
## [1] "Pruned Tree"
```



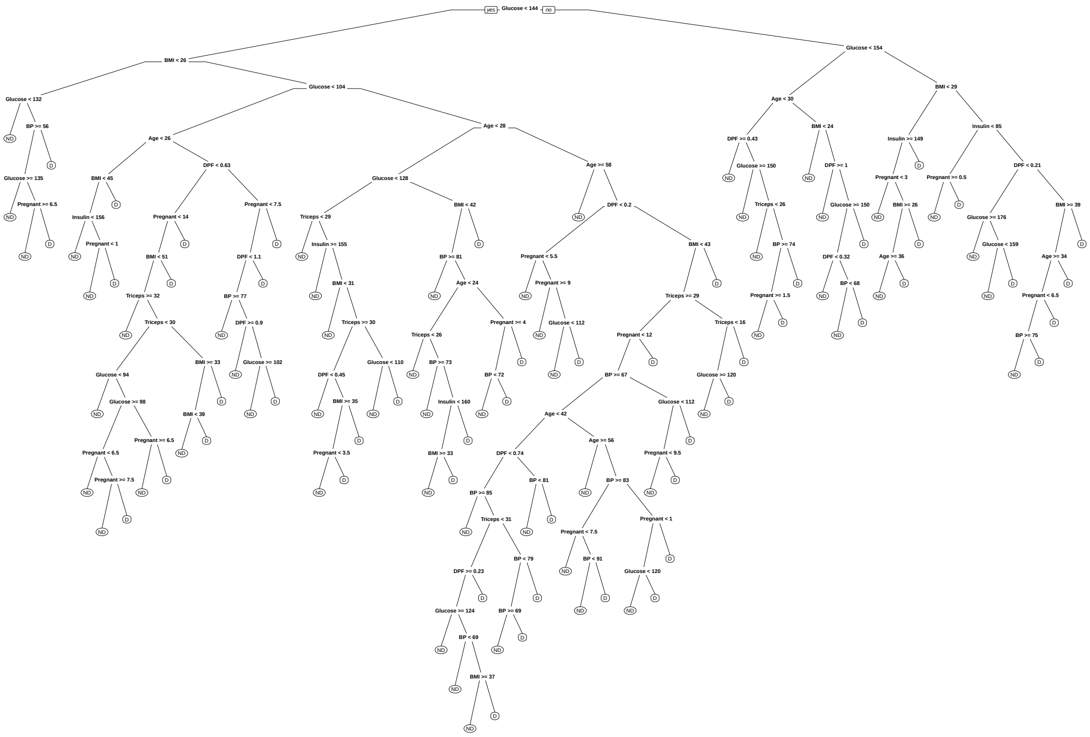
```
## [1] 0.751634
## [1] "Unpruned Tree"
```



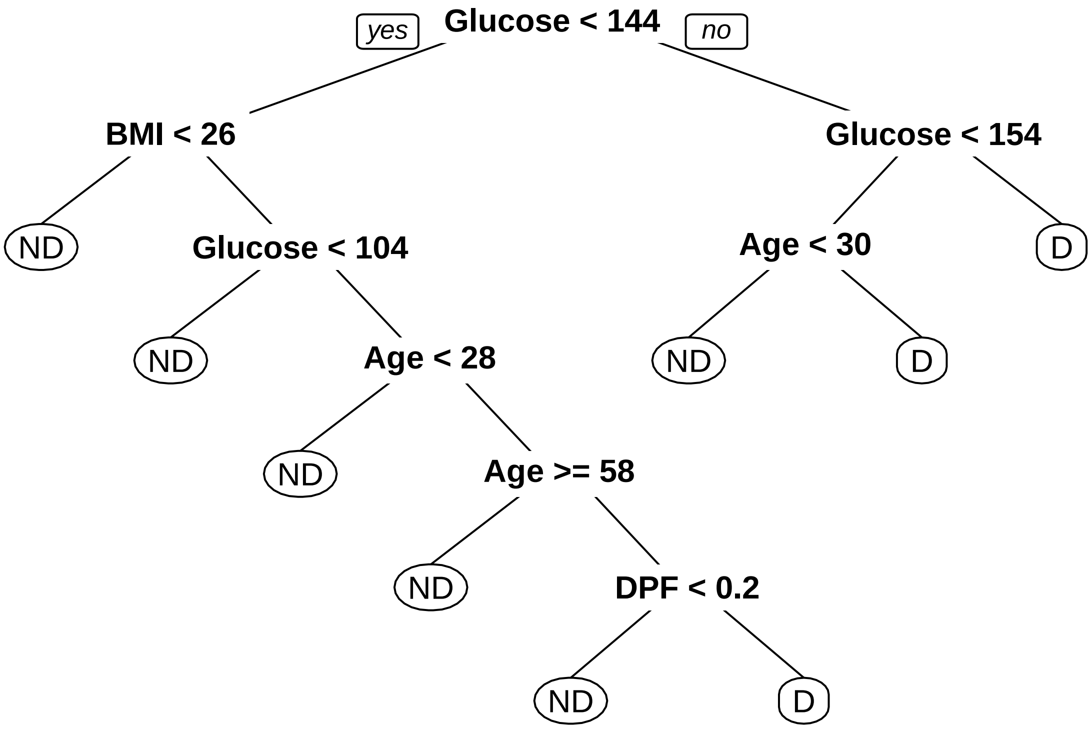
```
## [1] "Pruned Tree"
```



```
## [1] 0.745098
## [1] "Unpruned Tree"
```

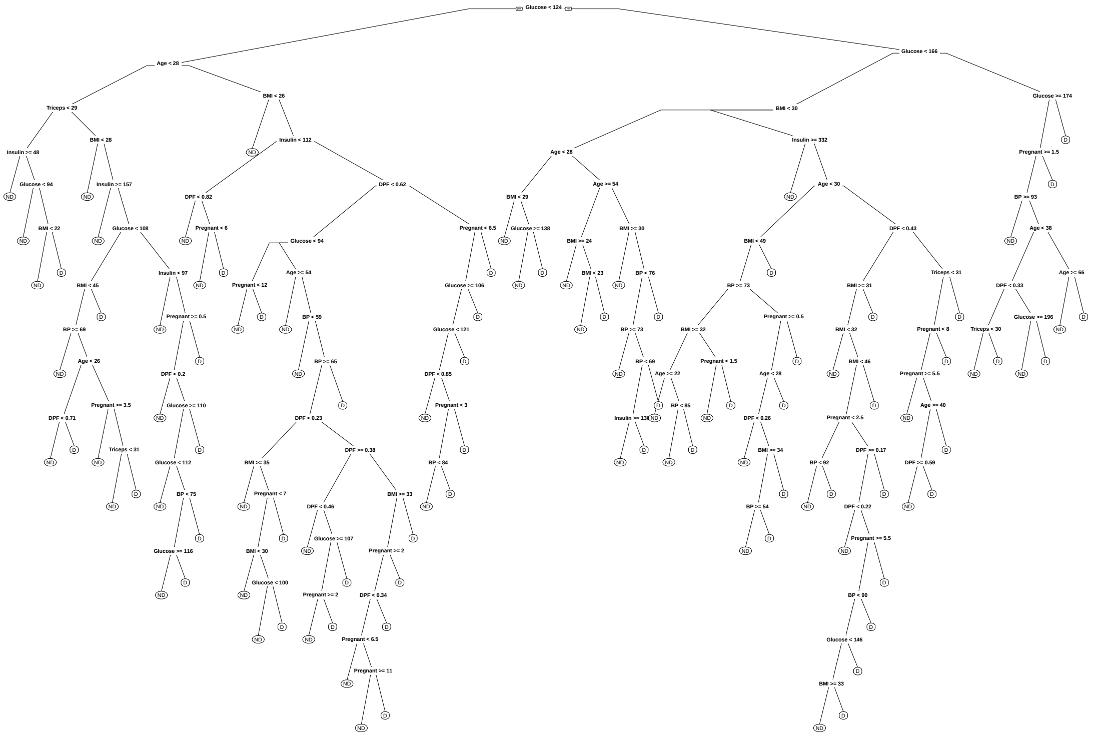


```
## [1] "Pruned Tree"
```

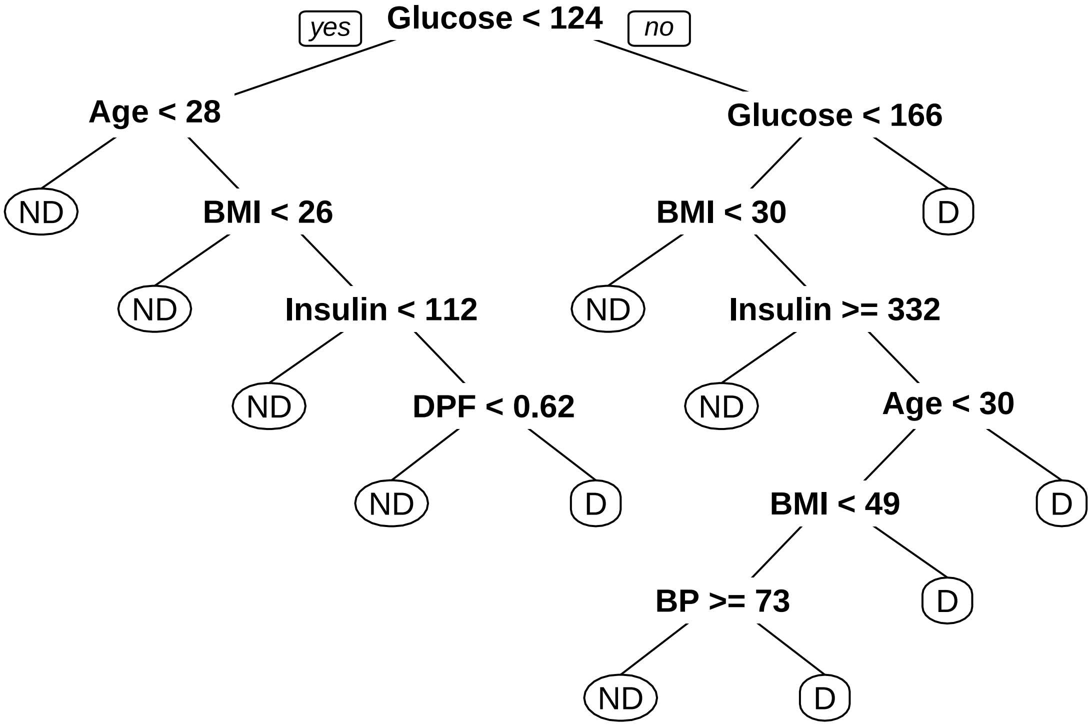


```
## [1] 0.7647059
## [1] "Unpruned Tree"
```

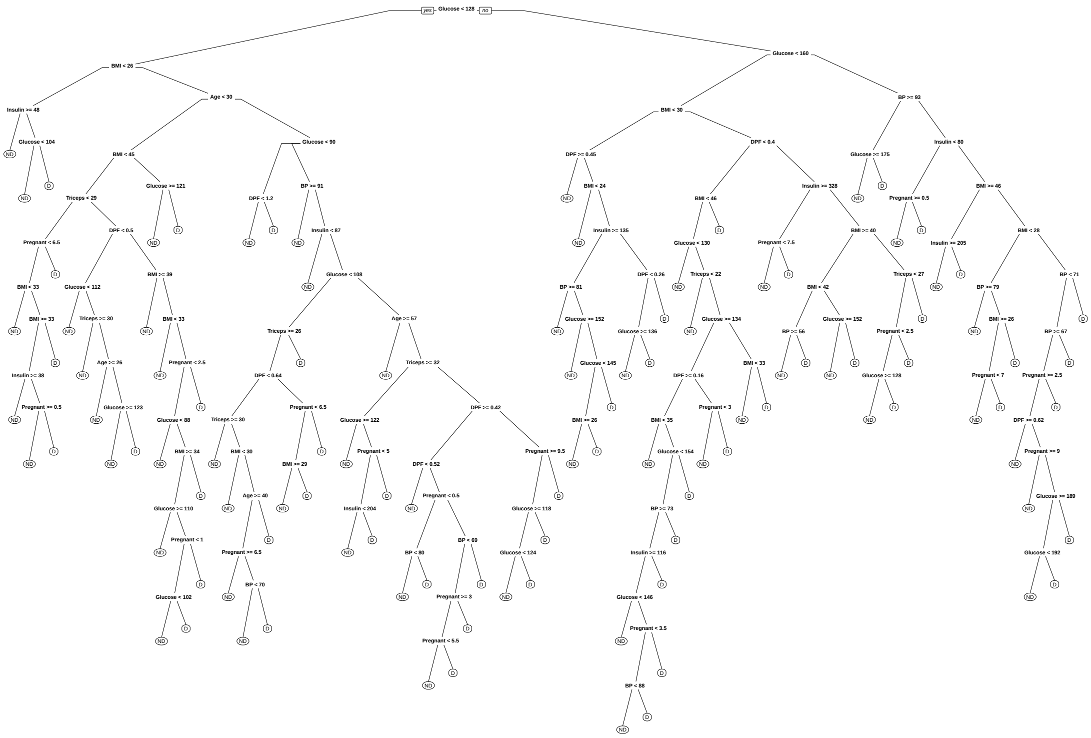
```
## Warning: labs do not fit even at cex 0.15, there may be some overplotting
```



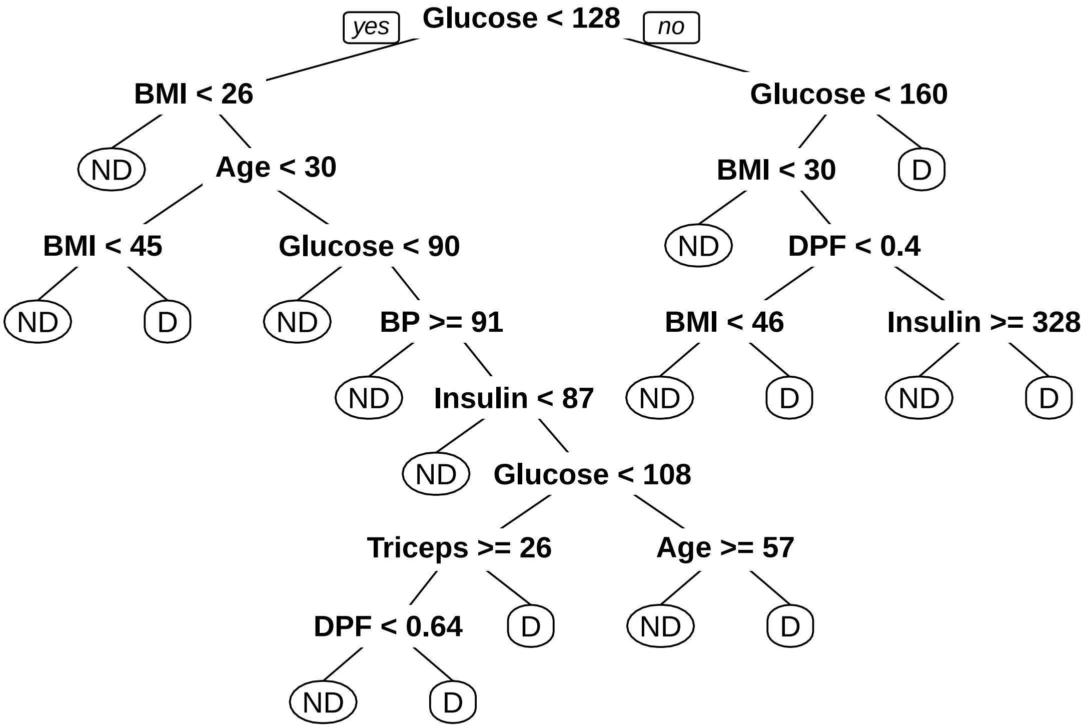
```
## [1] "Pruned Tree"
```



```
## [1] 0.7948718
## [1] "Unpruned Tree"
```



```
## [1] "Pruned Tree"
```



```r
#Mean Accuracy
print(Accuracy/K)
```

```
## [1] 0.7628959
```