DMPM LAB 8

Name: Saniya S. Inamdar

SRN: 201900913

Roll no. : 17

CODE:

```
library(tree)
library(rpart)
library(rpart.plot)
library(vip)
library(Metrics)
prostate = read.csv("D:/TY sem6/DMPM LAB/Assn6/prostate.csv")

head(prostate)
str(prostate)
dim(prostate) #97x6
summary(prostate)
#no NANs, need to scale the data
#lcavol - response variable


#split
set.seed(123)
sample_ind = sample(nrow(prostate),nrow(prostate)*0.80)
train = prostate[sample_ind,]
test = prostate[-sample_ind,]

dim(train)
dim(test)


pstree <- rpart(
  formula = lcavol ~ .,
  data    = train,
  method  = "anova"
  , control = list(cp = 0, maxdepth = 30,minsplit = 20)
)
rpart.plot(pstree)
plotcp(pstree)
preds = predict(pstree, test)

cat("RMSE: ", rmse(test$lcavol,preds),"\nMAE: ",
mae(test$lcavol,preds),
    "\nMSE: ", mse(test$lcavol,preds))
```

```
rpart.plot(pstree)
plotcp(pstree)

printcp(pstree)

vip(pstree, num_features = 5)

#pruning

prunedTree <- rpart(
  formula = lcavol ~ .,
  data    = train,
  method  = "anova"
  , control = list(cp = 0.01)
)
preds2 = predict(prunedTree, test)

cat("RMSE: ", rmse(test$lcavol,preds2),"\nMAE: ",
mae(test$lcavol,preds2),
    "\nMSE: ", mse(test$lcavol,preds2))
rpart.plot(prunedTree)
plotcp(prunedTree)
```

OUTPUT:

```
> head(prostate)
      lcavol age      lbph       lcp gleason       lpsa
1 -0.5798185  50 -1.386294 -1.386294       6 -0.4307829
2 -0.9942523  58 -1.386294 -1.386294       6 -0.1625189
3 -0.5108256  74 -1.386294 -1.386294       7 -0.1625189
4 -1.2039728  58 -1.386294 -1.386294       6 -0.1625189
5  0.7514161  62 -1.386294 -1.386294       6  0.3715636
6 -1.0498221  50 -1.386294 -1.386294       6  0.7654678
> str(prostate)
'data.frame':   97 obs. of  6 variables:
 $ lcavol : num  -0.58 -0.994 -0.511 -1.204 0.751 ...
 $ age    : int  50 58 74 58 62 50 64 58 47 63 ...
 $ lbph   : num  -1.39 -1.39 -1.39 -1.39 -1.39 ...
 $ lcp    : num  -1.39 -1.39 -1.39 -1.39 -1.39 ...
 $ gleason: int  6 6 7 6 6 6 6 6 6 6 ...
 $ lpsa   : num  -0.431 -0.163 -0.163 -0.163 0.372 ...
> dim(prostate) #97x6
[1] 97  6
> summary(prostate)
     lcavol              age             lbph                lcp
 Min.   :-1.3471   Min.   :41.00   Min.   :-1.3863   Min.   :-1.3863
 1st Qu.: 0.5128   1st Qu.:60.00   1st Qu.:-1.3863   1st Qu.:-1.3863
 Median : 1.4469   Median :65.00   Median : 0.3001   Median :-0.7985
 Mean   : 1.3500   Mean   :63.87   Mean   : 0.1004   Mean   :-0.1794
 3rd Qu.: 2.1270   3rd Qu.:68.00   3rd Qu.: 1.5581   3rd Qu.: 1.1787
 Max.   : 3.8210   Max.   :79.00   Max.   : 2.3263   Max.   : 2.9042
    gleason            lpsa
 Min.   :6.000   Min.   :-0.4308
 1st Qu.:6.000   1st Qu.: 1.7317
 Median :7.000   Median : 2.5915
 Mean   :6.753   Mean   : 2.4784
 3rd Qu.:7.000   3rd Qu.: 3.0564
 Max.   :9.000   Max.   : 5.5829
>
```

There are total 6 features: lcavol is the response variable. The data is not scaled but that would make no difference to the model since the model is decision tree model.

There are no Empty or NaN or missing values in the dataset.

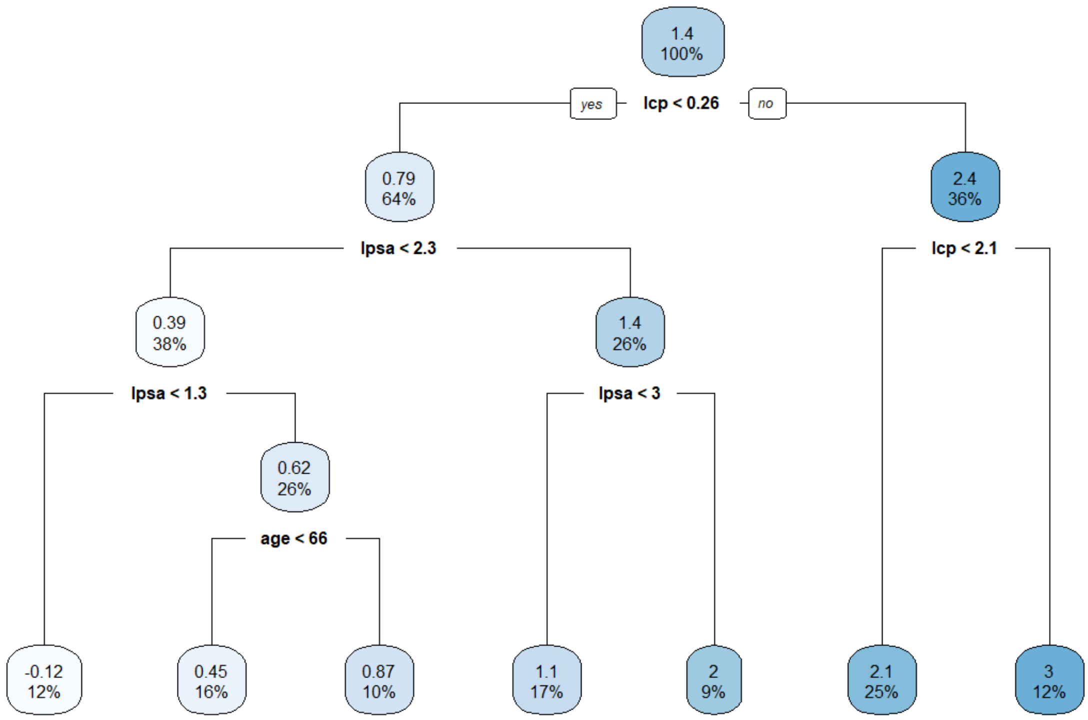
```
> dim(train)
[1] 77  6
> dim(test)
[1] 20  6
>
```

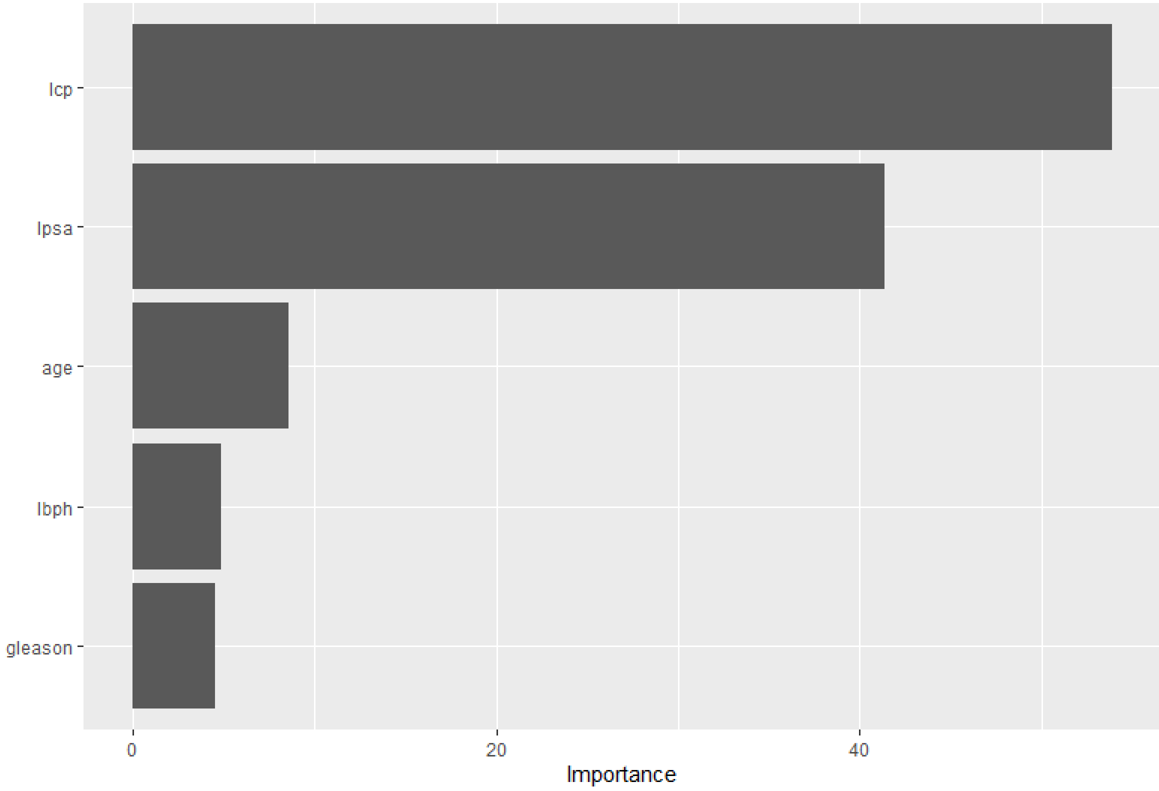
The data is split into training and test data using 80:20 ratio.

Creating the model:

<u>Before pruning:</u>

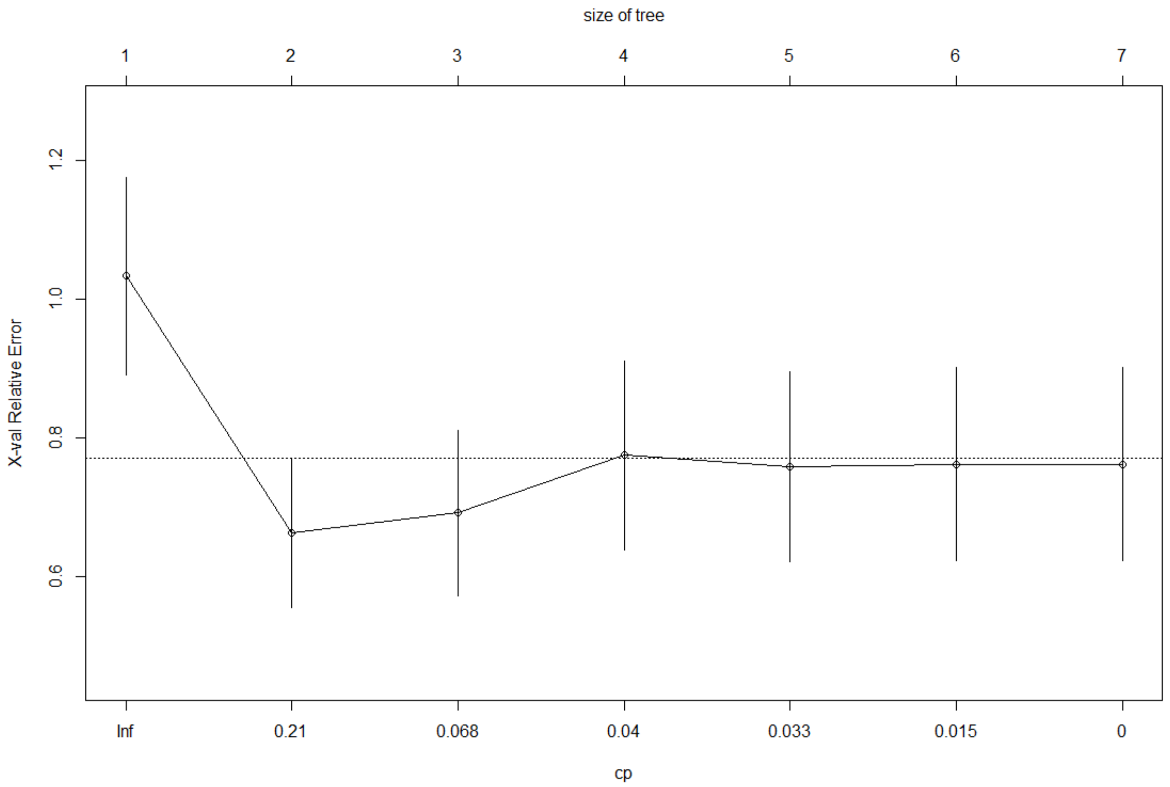
This is the base model, where the tree is allowed to grow, the following tree is formed:





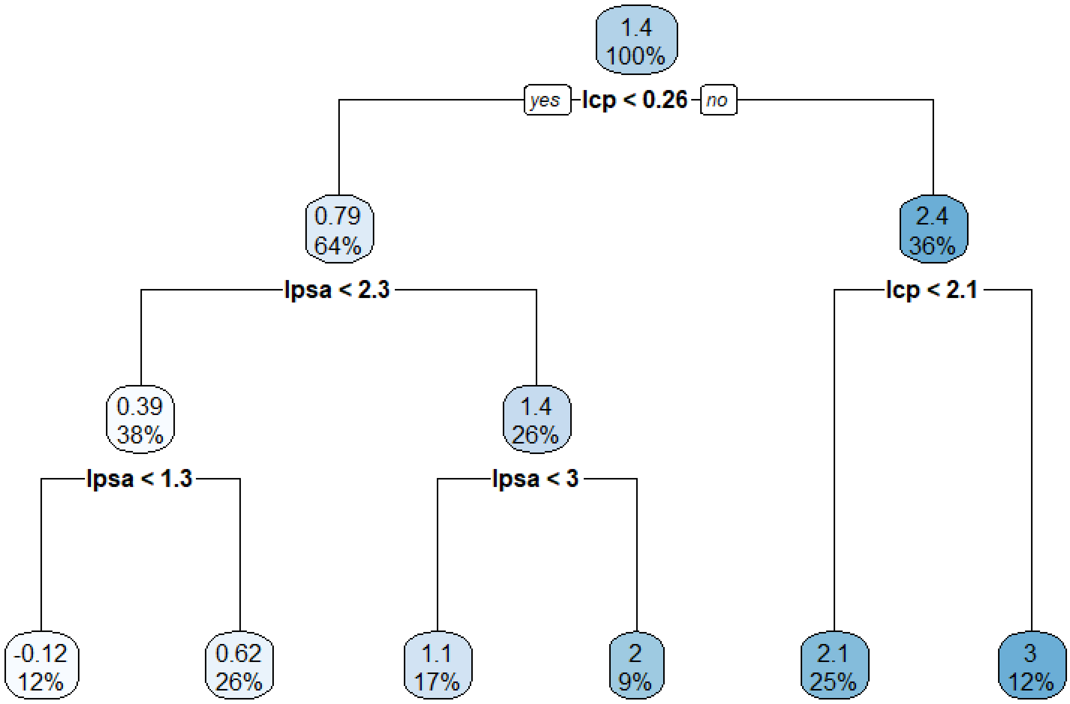
Here’s how the features are given importance by the tree. The most important features are **lcp and lpsa**. Dropping the rest of the features makes no change in the metrics.

```
> printcp(pstree)

Regression tree:
rpart(formula = lcavol ~ ., data = train, method = "anova", control = list(cp = 0,
    maxdepth = 30, minsplit = 20))

Variables actually used in tree construction:
[1] age  lcp  lpsa

Root node error: 110.35/77 = 1.4331

n= 77

         CP nsplit rel error  xerror    xstd
1 0.4333351      0   1.00000 1.03268 0.14164
2 0.1063154      1   0.56666 0.66221 0.10799
3 0.0440216      2   0.46035 0.69114 0.11905
4 0.0355252      3   0.41633 0.77413 0.13670
5 0.0307296      4   0.38080 0.75822 0.13727
6 0.0074876      5   0.35007 0.76152 0.13933
7 0.0000000      6   0.34259 0.76152 0.13933
> 
```



Above graph shows the size of the tree. From the above plot we can observe the cp value. The CP (complexity parameter) is used to control tree growth. If the cost of adding a variable is higher then the value of CP, then tree growth stops.

After pruning:
From the above observations, I have taken cp=0.01.



Results:

Before pruning:

```
RMSE:  0.6025333
MAE:  0.5214287
MSE:  0.3630464
```

After pruning:

```
RMSE:  0.5862111
MAE:  0.5172799
MSE:  0.3436435
```

I have also observed that if the ratio of training dataset is increased the decision tree performs better. So If a larger dataset was provided the performance would be much better.

END