Homework-5

Group 4

6/11/2020

Problem 1

The file ToyotaCorolla.xlsx contains the data on used cars (Toyota Corolla) on sale during late summer of 2004 in The Netherlands. It has 1436 records containing details on 38 attributes, including Price, Age, Kilometers, HP, and other specifications. The goal is to predict the price of a used Toyota Corolla based on its specifications.

Data Preprocessing: Create dummy variables for the categorical predictors (Fuel Type and Color). Split the data into training (50%), validation (30%), and test (20%) datasets.

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(readxl)
library(ggplot2)
library(caret)
```

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

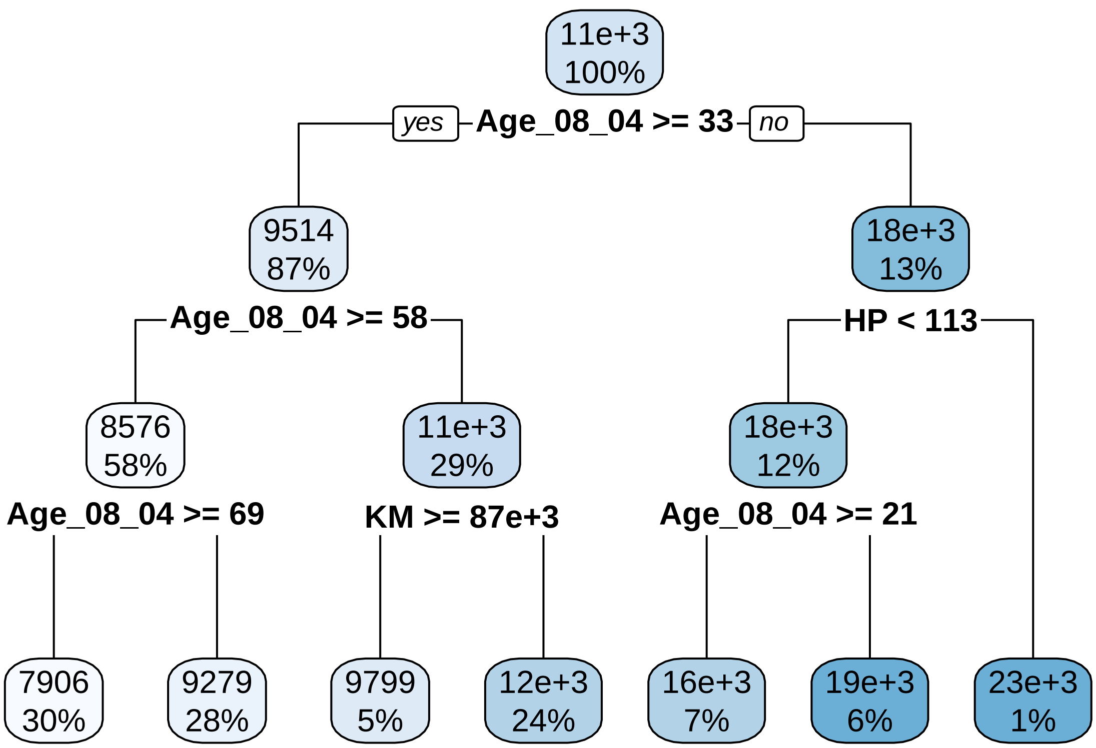
```r
library(tree)
library(rpart)
library(rpart.plot)
library(OneR)
library(reshape2)

ToyotaCorolla <- read_xlsx("ToyotaCorolla.xlsx", sheet = "data")

dummies <- dummyVars(~ Fuel_Type + Color, data = ToyotaCorolla, sep = ".")
dummies <-predict(dummies, ToyotaCorolla)

ToyotaCorolla <- cbind(select(ToyotaCorolla, -c("Fuel_Type", "Color")),
                       as.data.frame(dummies))

set.seed(20)
split_sample <- sample(1:3,
```

```
                        size = nrow(ToyotaCorolla),
                        prob = c(0.50, 0.30, 0.20),
                        replace = TRUE)

train_data <- ToyotaCorolla[split_sample == 1, ]
valididation_data <- ToyotaCorolla[split_sample == 2, ]
test_data <- ToyotaCorolla[split_sample == 3, ]
```

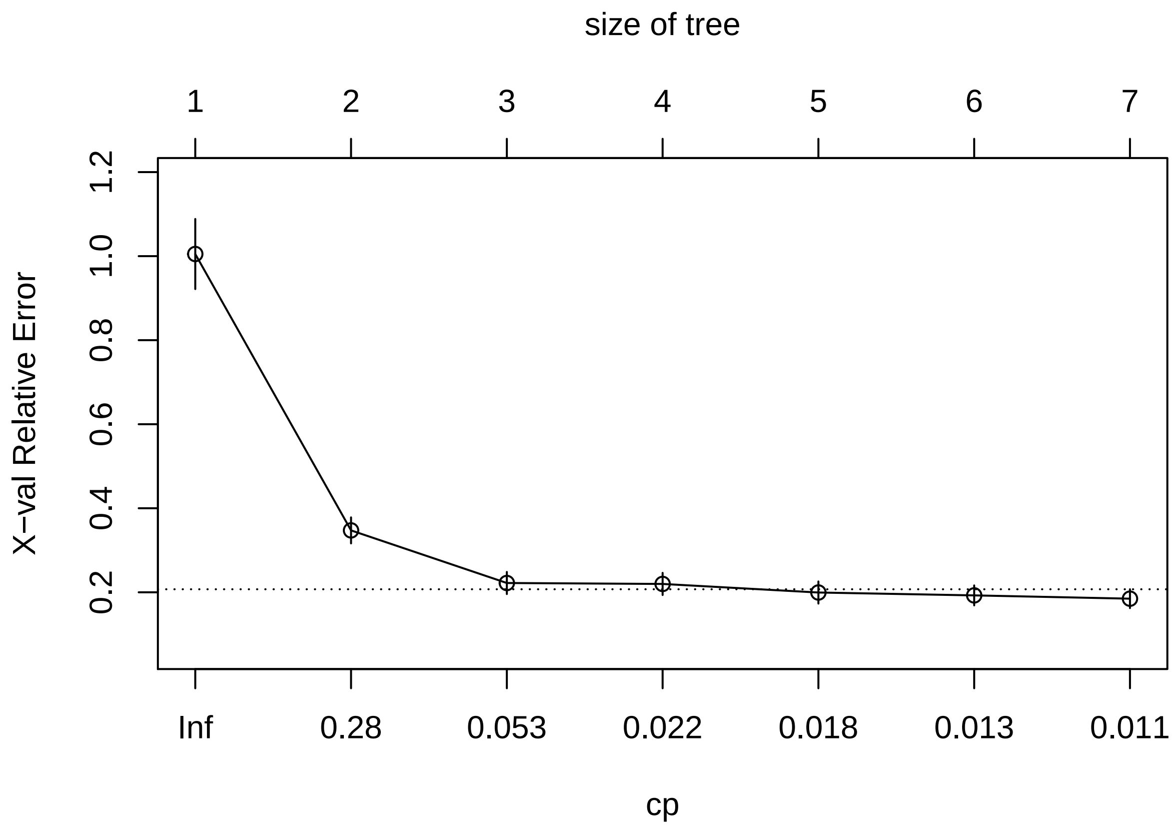
(a) Run a regression tree (RT) with the output variable Price and input variables

Age_08_04, KM, Fuel_Type, HP, Automatic, Doors, Quarterly_Tax, Mfg_Guarantee, Guarantee_Period, Airco, Automatic_Airco, CD Player, Powered_Windows, Sport_Model, and Tow_Bar.

```
reg.tree <- rpart(Price ~ Age_08_04 + KM + Fuel_TypeCNG + Fuel_TypeDiesel +
                      Fuel_TypePetrol + HP + Automatic + Doors +
                      Quarterly_Tax + Mfr_Guarantee + Guarantee_Period + Airco +
                      Tow_Bar,
              data = train_data,
              method = "anova")

rpart.plot(reg.tree)
```




```
plotcp(reg.tree)
```




```r
print(reg.tree$variable.importance)
```

```
##        Age_08_04  Automatic_airco               KM    Quarterly_Tax
##       7716864493       2456255799       1812651349        837860826
##               HP Guarantee_Period        CD_Player  Fuel_TypeDiesel
##        811189489        336999867        219398681         34185858
##  Fuel_TypePetrol            Doors            Airco  Powered_Windows
##         34185858         31068708         27966026         18322568
##    Mfr_Guarantee      Sport_Model
##         17358223          3107805
```

(i) Which appear to be the three or four most important car specifications for predicting the car's price?

- The most important car specifications for predicting the car's price are Age_08_04, KM and Automatic_airco.

(ii) Compare the prediction errors of the training, validation, and test sets by examining their RMS error and by plotting the three boxplots. What is happening with the training set predictions? How does the predictive performance of the test set compare to the other two? Why does this occur?

```r
train_preds <- predict(reg.tree, train_data)
validation_preds <- predict(reg.tree, valididation_data)
test_preds <- predict(reg.tree, test_data)

train_error <- RMSE(train_preds, train_data$Price)
validation_error <- RMSE(validation_preds, valididation_data$Price)
test_error <- RMSE(test_preds, test_data$Price)
```

```r
cat("Train Data RMSE", train_error, "\n")
```

```
## Train Data RMSE 1345.008
```

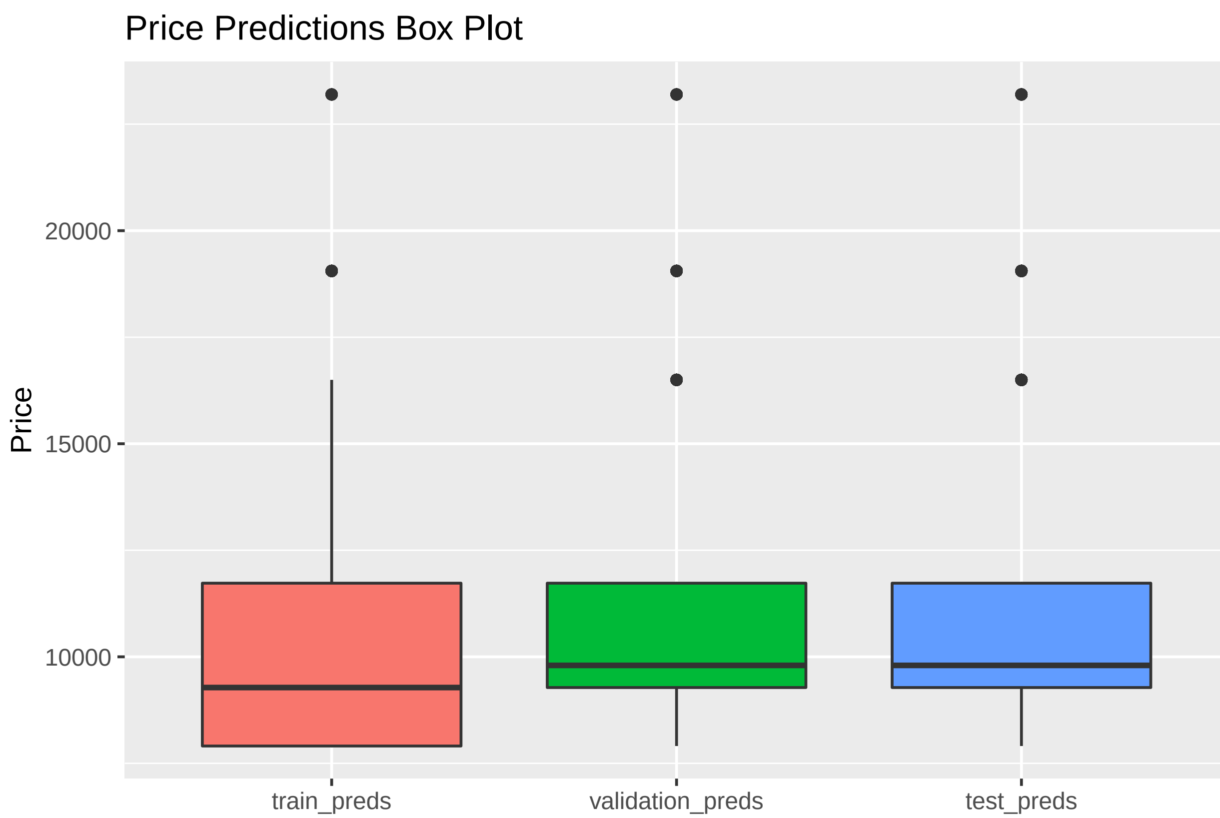
```r
cat("Validation Data RMSE", validation_error, "\n")
```

```
## Validation Data RMSE 1397.415
```

```r
cat("Test Data RMSE", test_error, "\n")
```

```
## Test Data RMSE 1591.446
```

```r
df <- melt(as.data.frame(cbind(train_preds, validation_preds, test_preds)))
ggplot(data = df, aes(x = variable, y = value, fill = variable)) +
  geom_boxplot() +
  theme(legend.position = "none") +
  ggtitle("Price Predictions Box Plot") +
  labs(x = "", y = "Price")
```



We can see from the RMSE error that the model performs the best on the train_data but that is to be expected. It performs slightly worse on the validation data. It performs the worst on the test data.

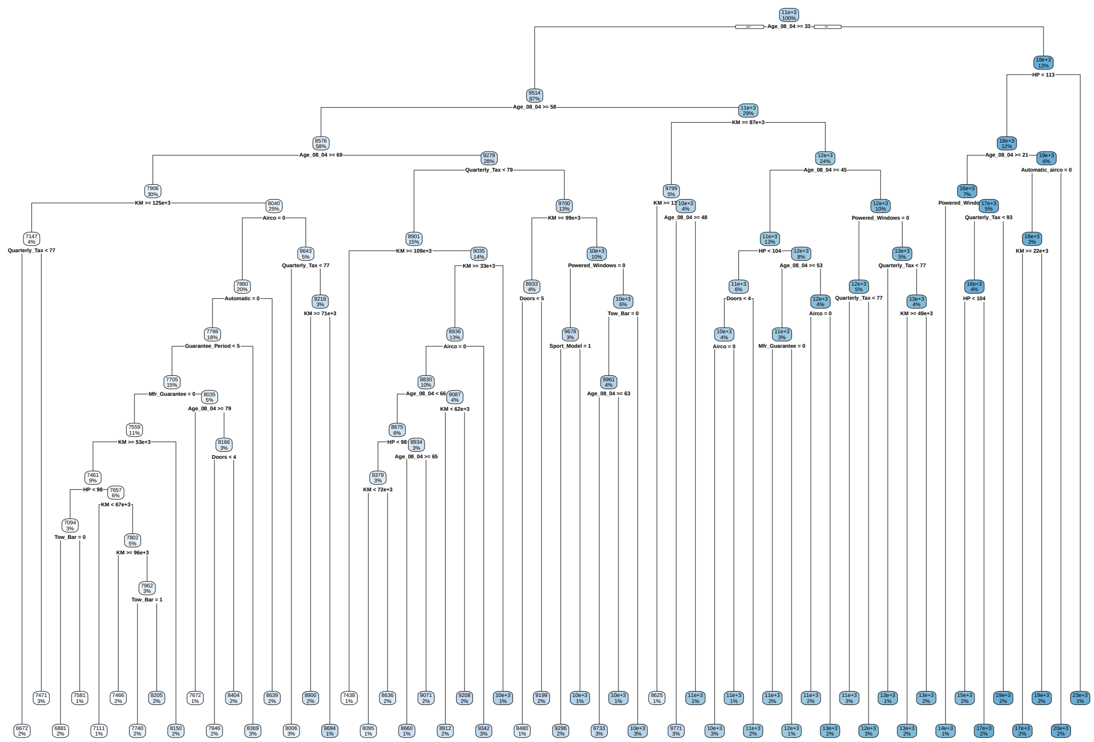
We can see that the training set predictions are similar to the actual Price values of the train data with a mean of approx $10000. There are 2 samples which are outliers in all three of the sets.

(iv) If we used the full tree instead of the best pruned tree to score the validation set, how would this affect the predictive performance for the validation set? (Hint: Does the full tree use the validation data?)

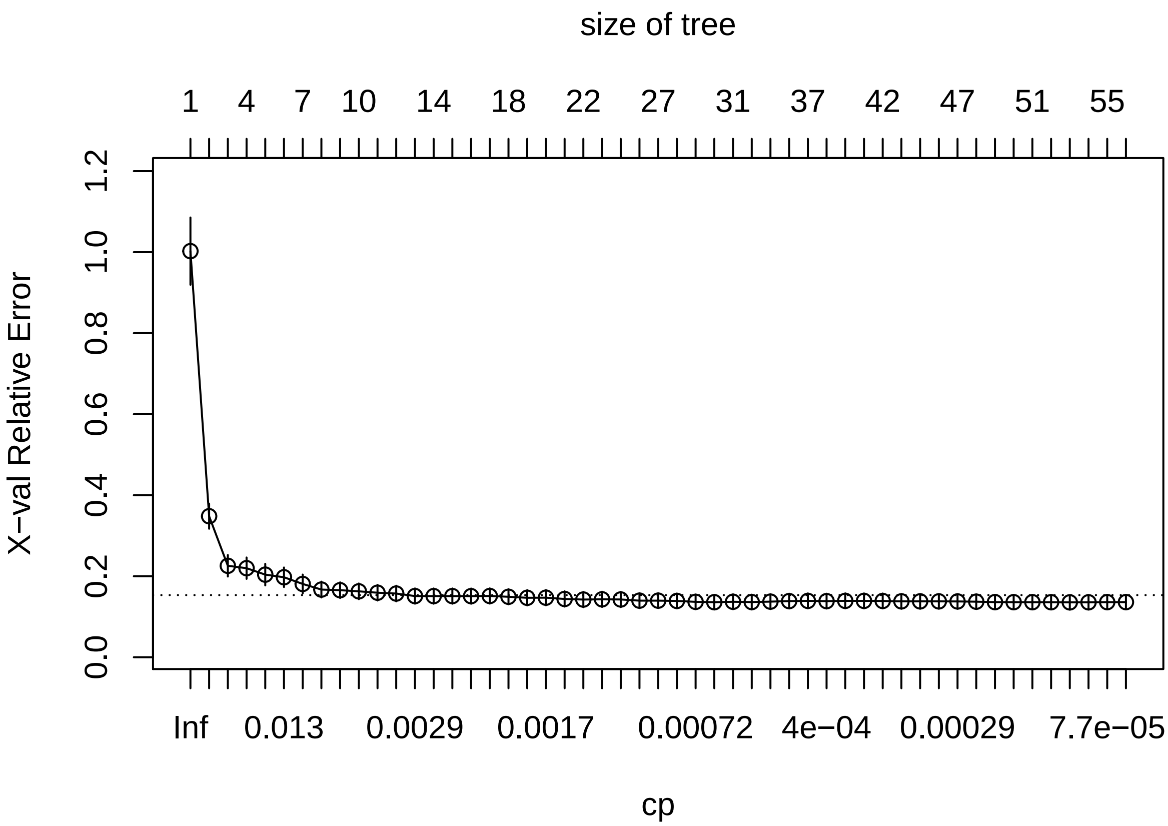
```r
full.tree <- rpart(Price ~ Age_08_04 + KM + Fuel_TypeCNG + Fuel_TypeDiesel +
                      Fuel_TypePetrol + HP + Automatic + Doors +
```

```r
                      Quarterly_Tax + Mfr_Guarantee + Guarantee_Period + Airco +
                      Tow_Bar,
              data = train_data,
              method = "anova",
              control = list(cp = 0))

rpart.plot(full.tree)
```

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



```r
plotcp(full.tree)
```




```r
train_preds <- predict(full.tree, train_data)
validation_preds <- predict(full.tree, valididation_data)
test_preds <- predict(full.tree, test_data)

train_error <- RMSE(train_preds, train_data$Price)
validation_error <- RMSE(validation_preds, valididation_data$Price)
test_error <- RMSE(test_preds, test_data$Price)

cat("Train Data RMSE", train_error, "\n")
```

```
## Train Data RMSE 971.6847
```

```r
cat("Validation Data RMSE", validation_error, "\n")
```

```
## Validation Data RMSE 1202.67
```

```r
cat("Test Data RMSE", test_error, "\n")
```

```
## Test Data RMSE 1497.919
```

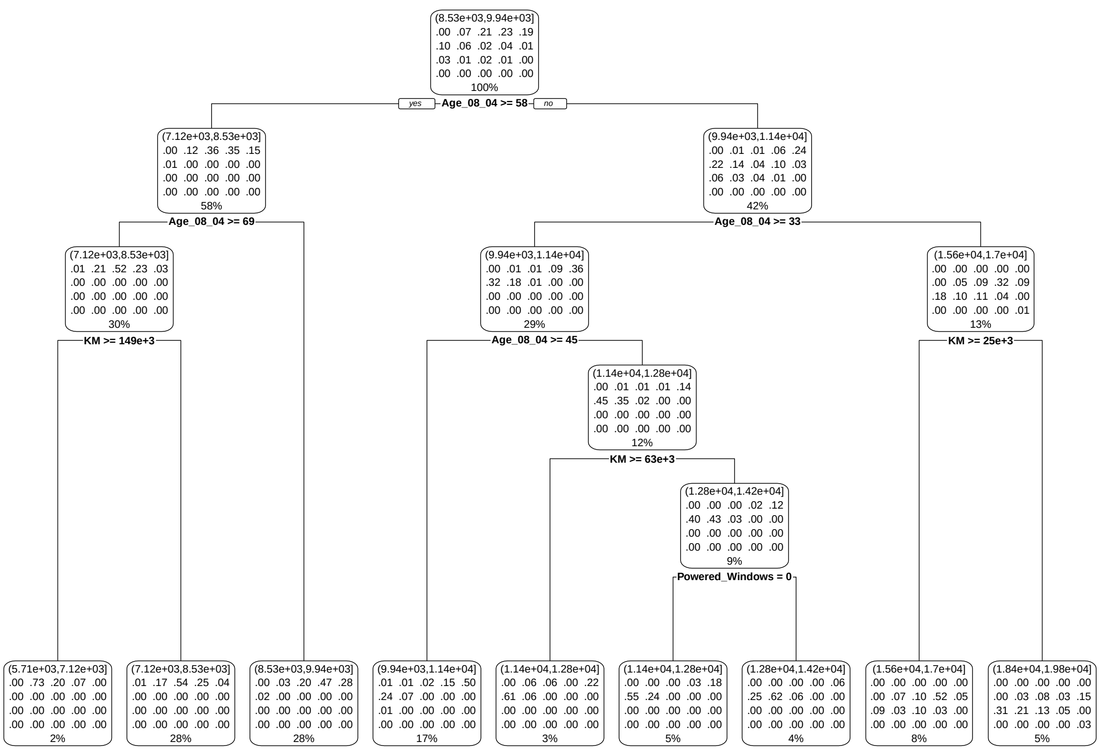
Using the full tree, we significantly reduce the RMSE error on both the validation and the test sets. Although the full tree doesn't use the validation data, it has more decision boundaries which are correctly separating the validation data hence reducing its error.

(b) Let us see the effect of turning the price variable into a categorical variable. First, create a new variable that categorizes price into 20 bins of equal counts. Now repartition the data keeping Binned Price instead of Price. Run a classification tree (CT) with the same set of input variables as in the RT, and with Binned Price as the output variable.

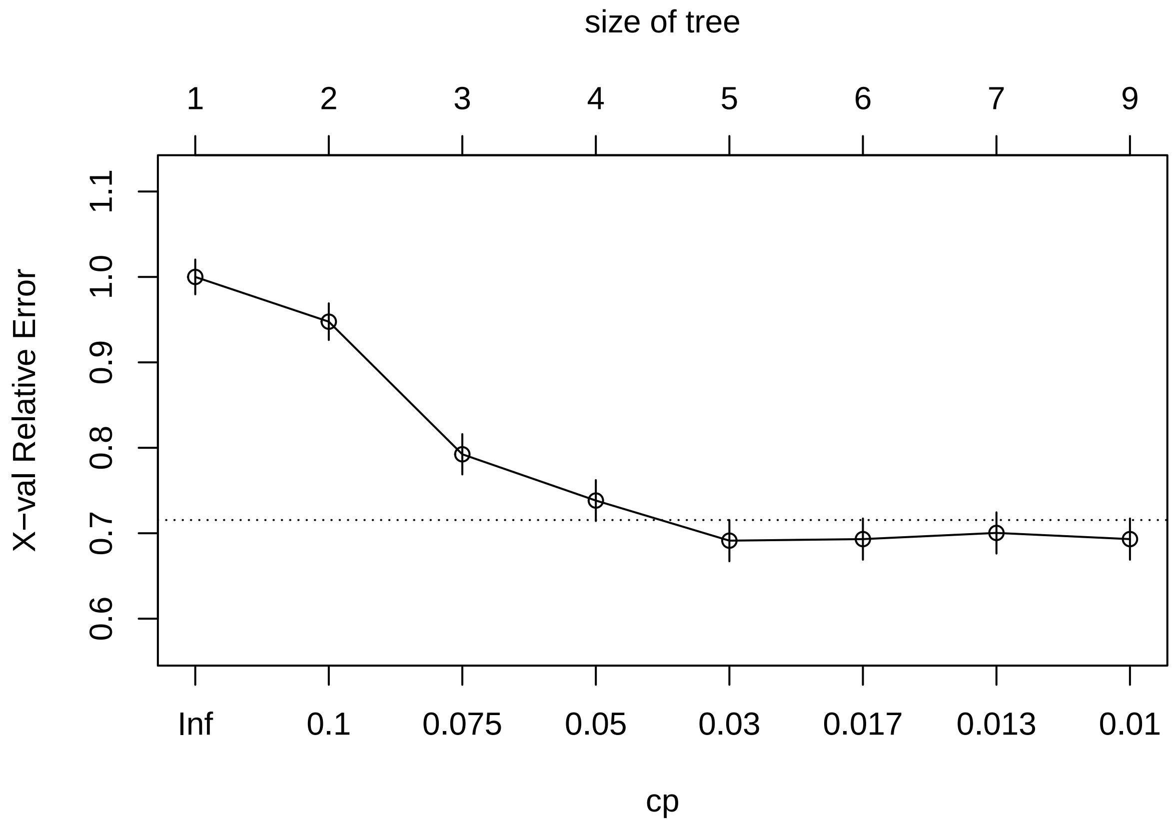
```r
ToyotaCorolla$Price <- cut(ToyotaCorolla$Price, breaks = seq(4300, 32500, by = 1410))

set.seed(20)
split_sample <- sample(1:3,
                       size = nrow(ToyotaCorolla),
                       prob = c(0.50, 0.30, 0.20),
                       replace = TRUE)

train_data <- ToyotaCorolla[split_sample == 1, ]
valididdation_data <- ToyotaCorolla[split_sample == 2, ]
test_data <- ToyotaCorolla[split_sample == 3, ]
```

```r
class.tree <- rpart(Price ~ Age_08_04 + KM + Fuel_TypeCNG + Fuel_TypeDiesel +
                     Fuel_TypePetrol + HP + Automatic + Doors +
                     Quarterly_Tax + Mfr_Guarantee + Guarantee_Period + Airco +
                     Tow_Bar,
               data = train_data)

rpart.plot(class.tree)
```

```
## Warning: All boxes will be white (the box.palette argument will be ignored) because
## the number of classes in the response 20 is greater than length(box.palette) 6.
## To silence this warning use box.palette=0 or trace=-1.
```



```r
plotcp(class.tree)
```




```r
print(class.tree$variable.importance)
```

```
##         Age_08_04                KM             Airco    Automatic_airco
##       127.3907916        53.2246381        19.4035608         18.5668915
##         CD_Player       Sport_Model     Quarterly_Tax      Mfr_Guarantee
##        18.2603264        15.7928942         7.3889731          6.7235932
##    Powered_Windows                HP  Fuel_TypePetrol  Guarantee_Period
##         6.5573410         3.8118783         1.1929470          0.9997784
##             Doors  Fuel_TypeDiesel      Fuel_TypeCNG          Automatic
##         0.8981243         0.7550886         0.3205369          0.1602685
```

(i) Compare the tree generated by the CT with the one generated by the RT. Are they different? (Look at structure, the top predictors, size of tree, etc.) Why?

- The top predictors of both the classification and regression trees remain more or less the same except of Quarterly_Tax which is only important in the regression tree.
- The classification tree is both structurally more complex as well bigger than the regression tree. This difference is created because in a regression tree, the response value is the mean of all the other values in the particular region however in the classification tree, the response is the majority class of the region.
- The difference in shape is so apparent because decision trees are sensitive to change in training data.

(ii) Predict the price, using the RT and the CT, of a used Toyota Corolla with the specifications listed in Table below.

```r
df <- data.frame(Age_08_04 = 77,
                 KM = 117000,
                 Fuel_TypePetrol = 1,
                 Fuel_TypeDiesel = 0,
                 Fuel_TypeCNG = 0,
                 HP = 110,
                 Automatic = 0,
                 Doors = 5,
                 Quarterly_Tax = 100,
                 Mfr_Guarantee = 0,
                 Guarantee_Period = 3,
                 Airco = 1,
                 Automatic_airco = 0,
                 CD_Player = 0,
                 Powered_Windows = 0,
                 Sport_Model = 0,
                 Tow_Bar = 1)


df.pred.reg <- predict(reg.tree, df)
df.pred.class <- as.data.frame(predict(class.tree, df))
df.pred.class <- attributes(which.max(df.pred.class))$names

cat("Prediction of Regression Tree:", df.pred.reg, "\n")
```

```
## Prediction of Regression Tree: 7906.141
```

```r
cat("Prediction of Classification Tree:", df.pred.class, "\n")
```

```
## Prediction of Classification Tree: (7.12e+03,8.53e+03]
```

(iii) Compare the predictions in terms of the predictors that were used, the magnitude of the difference between the two predictions, and the advantages and disadvantages of the two methods.

- The Regression Tree predicts the new data Price to be $7906.141. While the classification tree predicts that the price will be in the category/bin/range $7120-$8530.
- The advantage of using regression trees is that we can obtain an exact numerical prediction of our data.
- However classification trees can prove to be more useful in predicting numerical data in situations where we are more concerned with the range estimation of the data rather than it's average value.
