Decision Trees

Subramani.M

20 May 2018

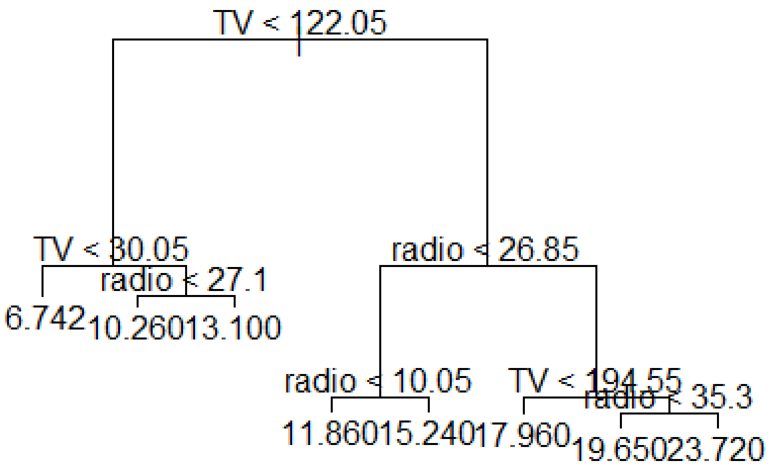
Decision trees with more than one input predictor (tv , radio , newspaper)

```
library(tree)

## Warning: package 'tree' was built under R version 3.4.4

advertising <- read.csv("C:/Users/Administrator/Desktop/Machine Learning/DATA
SETS/Advertising.csv")
model = tree(sales ~.,data = advertising)

{{plot(model)
  text(model)}}
```



```
# TV Cuts
TV_uniqs = sort(unique(advertising$TV))
length(TV_uniqs) # There were 10 duplicates which were removed. possible cuts
: 189

## [1] 190
```

```r
TV_uniqs[1:10]

##  [1]  0.7  4.1  5.4  7.3  7.8  8.4  8.6  8.7 11.7 13.1

cuts_Tv <- (TV_uniqs[1:length(TV_uniqs)-1] + TV_uniqs[2:length(TV_uniqs)]) /
2
length(cuts_Tv)

## [1] 189

# Radio cuts
radio_uniqs = sort(unique(advertising$radio))
length(radio_uniqs)

## [1] 167

radio_uniqs[1:10]

##  [1] 0.0 0.3 0.4 0.8 1.3 1.4 1.5 1.6 1.9 2.0

cuts_radio <- (radio_uniqs[1:length(radio_uniqs)-1] +
radio_uniqs[2:length(radio_uniqs)]) / 2
length(cuts_radio)

## [1] 166

# Newspapers cuts
np_uniqs = sort(unique(advertising$newspaper))
length(np_uniqs)

## [1] 172

np_uniqs[1:10]

##  [1] 0.3 0.9 1.0 1.7 1.8 2.1 2.2 2.4 3.2 3.6

cuts_np = (np_uniqs[1:length(np_uniqs)-1] + np_uniqs[2:length(np_uniqs)]) / 2
length(cuts_np)

## [1] 171

# method 1
library(dplyr)

## Warning: package 'dplyr' was built under R version 3.4.3

##
## 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
```

```
temp = advertising %>% filter(TV > 122.05 & radio > 26.85 & TV > 194.55) #
first value : TV < 122.05
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.3
```

```
tv_cuts_mse = c()

for(cut in cuts_Tv){
  samples_left = temp %>% filter(TV < cut)
  samples_right = temp %>% filter(TV > cut)
  pred_left = mean(samples_left$sales)
  pred_right = mean(samples_right$sales)
  temp$predict = ifelse(temp$TV < cut , pred_left , pred_right)
  curr_mse = sum((temp$sales - temp$pred) ^2)/nrow(temp)
  tv_cuts_mse = c(tv_cuts_mse,curr_mse)
}

radio_cuts_mse = c()
for(cut in cuts_radio){
  samples_left = temp %>% filter(radio < cut)
  samples_right = temp %>% filter(radio > cut)
  pred_left = mean(samples_left$sales)
  pred_right = mean(samples_right$sales)
  temp$predict = ifelse(temp$radio < cut , pred_left , pred_right)
  curr_mse = sum((temp$sales - temp$pred) ^2)/nrow(temp)
  radio_cuts_mse = c(radio_cuts_mse,curr_mse)
}

np_cuts_mse = c()
for(cut in cuts_np){
  samples_left = temp %>% filter(newspaper < cut)
  samples_right = temp %>% filter(newspaper > cut)
  pred_left = mean(samples_left$sales)
  pred_right = mean(samples_right$sales)
  temp$predict = ifelse(temp$newspaper < cut , pred_left , pred_right)
  curr_mse = sum((temp$sales - temp$pred) ^2)/nrow(temp)
  np_cuts_mse = c(np_cuts_mse,curr_mse)
}

result_TV = data.frame(column = rep('TV',length(cuts_Tv)), cut = cuts_Tv,mse
= tv_cuts_mse)
result_radio = data.frame(column = rep('radio',length(cuts_radio)), cut =
cuts_radio,mse = radio_cuts_mse)
result_np = data.frame(column = rep('newspaper',length(cuts_np)), cut =
cuts_np,mse = np_cuts_mse)
```

```
result = rbind(result_TV,result_radio,result_np)
View(result)
nrow(result)
```

```
## [1] 526
```

```
result %>% arrange(mse) %>% head(1) # Least mse . This becomes the parent
node
```

```
##   column   cut      mse
## 1  radio 34.45 1.782589
```

```
# method 2
cuts = c(cuts_Tv,cuts_radio,cuts_np)
predictors = c(rep('TV',length(cuts_Tv)),rep('radio',length(cuts_radio)),
                                           rep('newspaper',length(cuts_np)))
result = data.frame(cut = cuts,predictor = predictors)

cuts_mse = c()
var_devs <- c()
temp = advertising
for( i in seq(1,length(cuts))){
  cuts = cuts[i]
  curr_col = predictors[i]
  samples_left = temp[temp[,curr_col] < cut,]
  samples_right = temp[temp[,curr_col]>cut,]
  pred_left = mean(samples_left$sales)
  pred_right = mean(samples_right$sales)
  var_temp = var(temp$sales)
  var_left = var(samples_left$sales)
  var_right = var(samples_right$sales)
  var_Dev <- var_temp -(nrow(samples_left)/nrow(temp)*var_left)-
(nrow(samples_right)/nrow(temp)*var_right)
  temp$predict = ifelse(temp[,curr_col ]< cut, pred_left , pred_right)
  curr_mse = sum((temp$sales - temp$pred) ^2)/nrow(temp)
  cuts_mse = c(cuts_mse,curr_mse)
  var_devs = c(var_devs,var_Dev)
}
result$mse = cuts_mse
result$var_dev = var_devs
library(dplyr)
result %>% arrange(-var_dev) %>% head(1)
```

```
##   cut predictor      mse  var_dev
## 1 2.4        TV 14.30809 12.78475
```