Decision Tree Demo

Rob Carver

This document demonstrates (a) use of package party and well as (b) R Markdown to analyze data and create a presentation-ready document.

The example is adapted adapted from Shmueli, Patel, & Bruce, *Data Mining for Business Intelligence*, Chapter 9, wiley 2010

In a Markdown document, we insert chunks of R code between ordinary text. There are some important differences between writing typical scripts and using markdown. For example:

* by default, all code chunks display in the resulting document, as does output and error messages. We can suppress messages if we want.
* file paths must be typed out fully

Set your working directory before executing this script, and then we'll read the file with an explicit identification of the data directory.

library(party)

Next we read the data file for the problem. In this case, I first copied the "Mower.csv" file from GitHub and saved it in my data directory.

#Read file  
mydata<-read.csv("C:/Users/Rob/Box Sync/My R Work/BUS212/Data/Mower.csv",header=T)  
# header=T not needed,  
# but is a reminder that the option is available

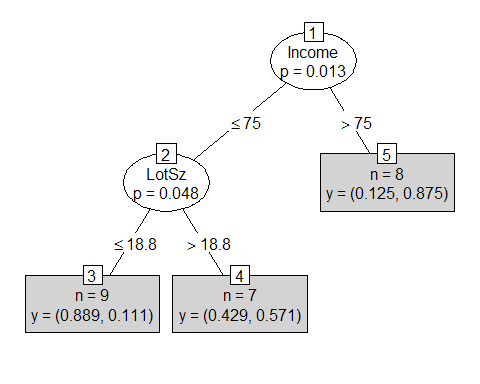
With the data in a dataframe, we can now build a tree using the ctree command in the party package. We identify the target (Y) variable and X factors to include in the tree.

The control options and theory are explained in class and in the assigned readings.

datactree <- ctree(Own~Income+LotSz, mydata,  
 controls=ctree\_control(mincriterion=0.9, minsplit=5))  
print(datactree) # print the tree rules

##   
## Conditional inference tree with 3 terminal nodes  
##   
## Response: Own   
## Inputs: Income, LotSz   
## Number of observations: 24   
##   
## 1) Income <= 75; criterion = 0.987, statistic = 7.463  
## 2) LotSz <= 18.8; criterion = 0.952, statistic = 5.085  
## 3)\* weights = 9   
## 2) LotSz > 18.8  
## 4)\* weights = 7   
## 1) Income > 75  
## 5)\* weights = 8

plot(datactree,type="simple") # display the tree



One simple way to evaluate the performance of a classification model is to create a cross-tab of the model decisions compared to the observed data. We call this cross-tab a *Confusion Matrix*, and compute the percentage of cases that were *misclassified* by the model.

# make the table and display it  
  
tab<-table(predict(datactree), mydata$Own)  
print(tab)

##   
## Non-Owner Owner  
## Non-Owner 8 1  
## Owner 4 11

# now compute misclassification rate  
1-sum(diag(tab))/sum(tab)

## [1] 0.2083333