Rpart Decision Tree Worksheet

# Single Decision Tree

## rpart package in R

Aim: Create a decision tree to predict income class of individuals found in the 'income' data set. This dataset contains information from 27,500 people including their income class, job, marital status and education level. Then, assess accuracy of predictions.

Note: ‘ \_\_\_\_\_\_\_\_\_\_?’ indicates task to complete or word to fill in

Nice link visualising a decision tree:

<http://www.r2d3.us/visual-intro-to-machine-learning-part-1/>

1. Open R Studio and set your working directory (where you have stored your datafile) using either setwd() or the 'Session' menu. For example:

setwd("C:/Users/Tim/OneDrive/SSA\_bigdata\_course/trees")

1. Install relevant packages: rpart, rpart.plot, ggplot2, GGally

install.packages('rpart')  
install.packages('rpart.plot')  
install.packages('ggplot2')  
install.packages('GGally')

1. Load packages:

library('rpart')  
library('rpart.plot')  
library('ggplot2')  
library('GGally')

1. Load 'income.csv' dataset and save as 'a'

a <- read.csv('income.csv')[,-1]

1. Explore dataset using commands such as summary(), str(), barplot(), plot(), table(), dim()

summary(a)  
str(a)  
dim(a)

#boxplots  
boxplot(age~class, a)  
ggplot(a, aes(x=class, y= age)) + geom\_boxplot()  
  
#grid of plots – data exploration  
ggpairs(a, columns = c(1,2,10))  
ggpairs(a, columns = c(3,4,10))  
ggpairs(a, columns = c(5,6,10))  
ggpairs(a, columns = c(7,8,10))  
ggpairs(a, columns = c(9,10))

Are there any NA values?

1. Split data into test and training sets

set.seed(100)  
sample<- sample(1:nrow(a), 2/3\*nrow(a), replace=F)  
train<- a[sample,]  
test<- a[-sample,]  
  
dim(train)

\_\_\_\_\_\_\_\_\_\_?

dim(test)

\_\_\_\_\_\_\_\_\_\_?

1. Create first tree, then prune using cp value as control.

fit<- rpart(class~., data=train, control = rpart.control(cp=0.0001))  
fit

printcp(fit)

plotcp(fit)

#prune to cp greater than \_\_\_\_\_\_\_\_\_?

fit<- rpart(class~., data=train, control = rpart.control(cp=\_\_\_\_\_\_\_?))  
fit

plotcp(fit)

* What other options are available in the rpart package under rpart.control?
* Optional: in the cp table, why does it skip more than one split in each step?

1. Plot the tree

plot(fit, margin=0.05, uniform = T)  
text(fit, use.n=T, cex=0.7)

Toy with the values in the functions

What does ‘cex’ do?

What does ‘uniform’ do?

1. Run train set through fitted tree. Exclude ‘class’ column

res.fit<- predict(fit, train[,-10])  
head(res.fit)

res.fit2<- predict(fit, train[,-10]), type = 'class')  
head(res.fit2)

1. Create data frame with train actual class 'answers' and train 'predicted' classes as the two columns

head(test)

# Make sure row numbers match

ans.fit<- data.frame('ans'=train$class, 'predict' = \_\_\_\_\_\_?)  
dim(ans.fit)

\_\_\_\_\_\_\_\_\_\_\_?

head(ans.fit)

1. Create confusion matrix for test set

table(ans.fit$ans, ans.fit$predict)

print out confusion matrix:

1. Repeat steps 9 through 11 with the test set instead of train set. This will give a more realistic confusion matrix, print confusion matrix below:
2. Create the tree plot from step 8 with interesting features or colours using the rpart.plot package. Flick through the document and choose a feature or two to try out:

<http://www.milbo.org/doc/prp.pdf>

1. Optional: try manual pruning found in rpart.plot

## 