Titanic - Recursive partitioning

### Variable Descriptions

**survival**: Survival (0 = No; 1 = Yes)

**pclass**: Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)

**name**: Name

**sex**: Sex

**age**: Age

**sibsp**: Number of Siblings/Spouses Aboard

**parch**: Number of Parents/Children Aboard

**ticket**: Ticket Number

**fare**: Passenger Fare

**cabin**: Cabin

**embarked**: Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

library(gtools)  
library(ggplot2)  
library(RCurl)  
library(rpart)  
library(rpart.plot)  
library(partykit)  
library(dplyr)

### 1. Reading data

url <- getURL('https://raw.githubusercontent.com/frankwwu/R-Knots/master/Titanic/train.csv')  
train <- read.csv(text = url)   
url <- getURL('https://raw.githubusercontent.com/frankwwu/R-Knots/master/Titanic/test.csv')  
test <- read.csv(text = url)

### 2. Removing names and NAs.

train<-train[, !(colnames(train) %in% c('name'))]  
train <-train %>% na.omit()  
dim(train)

## [1] 714 12

test<-test[, !(colnames(test) %in% c('name'))]  
test <- test %>% na.omit()  
dim(test)

## [1] 331 11

### 3. Selecting features

train$Survived <- factor(train$Survived)  
formula = Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked

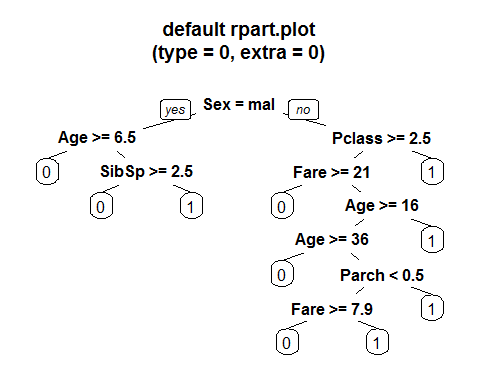
### 4. Classification Tree

Creating tree

fit <- rpart(formula, data=train, method="class")  
as.party(fit)

##   
## Model formula:  
## Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked  
##   
## Fitted party:  
## [1] root  
## | [2] Sex in male  
## | | [3] Age >= 6.5: 0 (n = 429, err = 17.9%)  
## | | [4] Age < 6.5  
## | | | [5] SibSp >= 2.5: 0 (n = 9, err = 11.1%)  
## | | | [6] SibSp < 2.5: 1 (n = 15, err = 0.0%)  
## | [7] Sex in female  
## | | [8] Pclass >= 2.5  
## | | | [9] Fare >= 20.8: 0 (n = 23, err = 13.0%)  
## | | | [10] Fare < 20.8  
## | | | | [11] Age >= 16.5  
## | | | | | [12] Age >= 36.5: 0 (n = 7, err = 14.3%)  
## | | | | | [13] Age < 36.5  
## | | | | | | [14] Parch < 0.5  
## | | | | | | | [15] Fare >= 7.8875: 0 (n = 23, err = 34.8%)  
## | | | | | | | [16] Fare < 7.8875: 1 (n = 16, err = 37.5%)  
## | | | | | | [17] Parch >= 0.5: 1 (n = 13, err = 30.8%)  
## | | | | [18] Age < 16.5: 1 (n = 20, err = 20.0%)  
## | | [19] Pclass < 2.5: 1 (n = 159, err = 5.7%)  
##   
## Number of inner nodes: 9  
## Number of terminal nodes: 10

rpart.plot(fit, main="default rpart.plot\n(type = 0, extra = 0)")

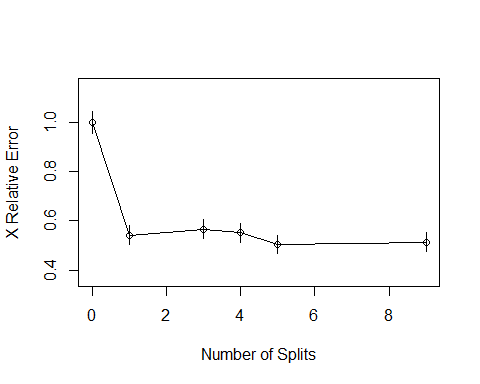
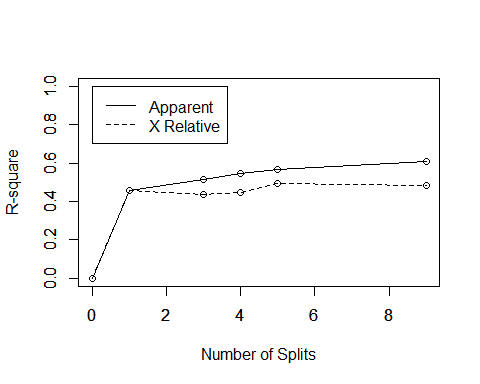


### 5. Validation

# Visualize cross-validation results  
rsq.rpart(fit)

##   
## Classification tree:  
## rpart(formula = formula, data = train, method = "class")  
##   
## Variables actually used in tree construction:  
## [1] Age Fare Parch Pclass Sex SibSp   
##   
## Root node error: 290/714 = 0.40616  
##   
## n= 714   
##   
## CP nsplit rel error xerror xstd  
## 1 0.458621 0 1.00000 1.00000 0.045252  
## 2 0.029310 1 0.54138 0.54138 0.038162  
## 3 0.027586 3 0.48276 0.56552 0.038758  
## 4 0.024138 4 0.45517 0.55172 0.038421  
## 5 0.010345 5 0.43103 0.50345 0.037162  
## 6 0.010000 9 0.38966 0.51379 0.037443

## Warning in rsq.rpart(fit): may not be applicable for this method



### 5. Prediction

predict <- predict(fit, test, type = "prob")