Introduction to `estimate.rpart`

Yanyang Kong

August 29, 2015

1 Usage

`estimate.rpart()` is built in package to estimate the causal effects of new observations using "honest" tree model, i.e. estimated value are computed from current observations but following the model's splitting rules.

To use `estimate.rpart()`, you should first build a causal tree model and then prune it. The pruned tree model should be inputed as an argument in our estimate function as below:

```
estimate.rpart(tree, y~., data = test_data, weights = test_wt)
```

Notice in our estimate function, if the leaf of the tree model contains only treated cases or control cases of new observations. The function will trace back to its parent node recursively until the parent node contains both cases and can be used to compute the causal effect:

$$\hat{\tau} = \bar{y}(w = 1) - \bar{y}(w = 0)$$

2 Example

Here is an example of how to use `estimate.rpart()`:

```
### estimate.rpart is to estimate causal effects of test data using honest tree:
library(rpart)
library(rpart.plot)
## generate data
x1 <- rnorm(1000, 0, 1)
x2 <- rnorm(1000, 0, 1)
x3 <- rnorm(1000, 0, 1)
x4 <- rnorm(1000, 0, 1)
x5 <- rnorm(1000, 0, 1)
x6 <- rnorm(1000, 0, 1)
x7 <- rnorm(1000, 0, 1)
x8 <- rnorm(1000, 0, 1)
x9 <- rnorm(1000, 0, 1)
x10 <- rnorm(1000, 0, 1)
w <- rep(1:0, each = 250)

## training 500:
```

```
y1_tr <- rnorm(250, 1 - x1[1:250] + x2[1:250], 1)
y0_tr <- rnorm(250, 0 + x1[251:500] + x2[251:500], 1)
y_tr <- c(y1_tr, y0_tr)
tr1 <- data.frame(x1[1:500], x2[1:500], x3[1:500], x4[1:500], x5[1:500], x6[1:500],
                  x7[1:500], x8[1:500], x9[1:500], x10[1:500], y_tr)
names(tr1) <- c("x1", "x2", "x3", "x4", "x5", "x6", "x7", "x8", "x9", "x10", "y")

## test 500:
y1_te <- rnorm(250, 1 - x1[501:750] + x2[501:750], 1)
y0_te <- rnorm(250, 0 + x1[751:1000] + x2[751:1000], 1)
y_te <- c(y1_te, y0_te)
te1 <- data.frame(x1[501:1000], x2[501:1000], x3[501:1000], x4[501:1000], x5[501:1000], x6[501:10
                  x7[501:1000], x8[501:1000], x9[501:1000], x10[501:1000], y_te)
names(te1) <- c("x1", "x2", "x3", "x4", "x5", "x6", "x7", "x8", "x9", "x10", "y")

## build and prune tree
tree1 <- rpart(y~., data = tr1, weights = w, method = "anova",
               cp = 0, parms = 1, minbucket = 1, cv.option = "matching")
opcp1 <- tree1$cp[which(tree1$cp[,4] == min(tree1$cp[,4])), 1]
prune1 <- prune(tree1, cp = opcp1)
rpart.plot(prune1)

## estimate the test data set:
est1 <- estimate.rpart(prune1, y~., data = te1, weights = w)
```