

policy_tree

2024-04-07

```
library(grf)
```

```
## Warning:  编辑包'grf'是用R版本4.3.3 来建造的
```

```
# library(policytree)
library(constrainedpolicytree)
library(policytree)
```

```
## Warning:  编辑包'policytree'是用R版本4.3.3 来建造的
```

```
## Registered S3 methods overwritten by 'policytree':
##   method          from
##   plot.policy_tree constrainedpolicytree
##   predict.policy_tree constrainedpolicytree
##   print.policy_tree  constrainedpolicytree
```

```
##
## 载入编辑包: 'policytree'
```

```
## The following object is masked from 'package:constrainedpolicytree':
##
##   policy_tree
```

```
# read inspire_CTS data
cts_data <- read.csv("inspire_CTS_0_120_icuhalfday.csv")
# sample size
sample_size <- 858
# covariate X, treatments W, outcome Y
X <- cts_data[,4:40]
#W <- as.factor(cts_data[,46])
W <- cts_data[,41]
Y <- cts_data[,1]
# translation of Y so that all Y are positive
C <- max(Y)+1
Y <- C-Y
# read IPW trained by sklearn.linear_model
IPW <- read.csv("IPW.csv")
# DR <- read.csv("DR.csv")
#c.forest <- grf::multi_arm_causal_forest(X, Y, W)
#dr.scores <- double_robust_scores(c.forest)
#write.csv(dr.scores, file = "dr_scores.csv", row.names = FALSE)
# train the policy tree
tree_cts <- constrainedpolicytree::policy_tree(X, IPW, max_treatment_size = c(344,172,258,858),
depth = 1)
```

```
## The values of max_treatment_size_vector: 344 172 258 858
```

```
# tree_cts <- policytree::policy_tree(X, DR, depth = 1)
print(tree_cts)
```

```
## policy_tree object
## Tree depth: 1
## Actions: 1: X0 2: X1 3: X2 4: X3
## Variable splits:
## (1) split_variable: alt split_value: 22
## (2) * action: 4
## (3) * action: 3
```

```
print(tree_cts$`_tree_array`)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]  12  22   2   3
## [2,]  -1   4   0   0
## [3,]  -1   3   0   0
```

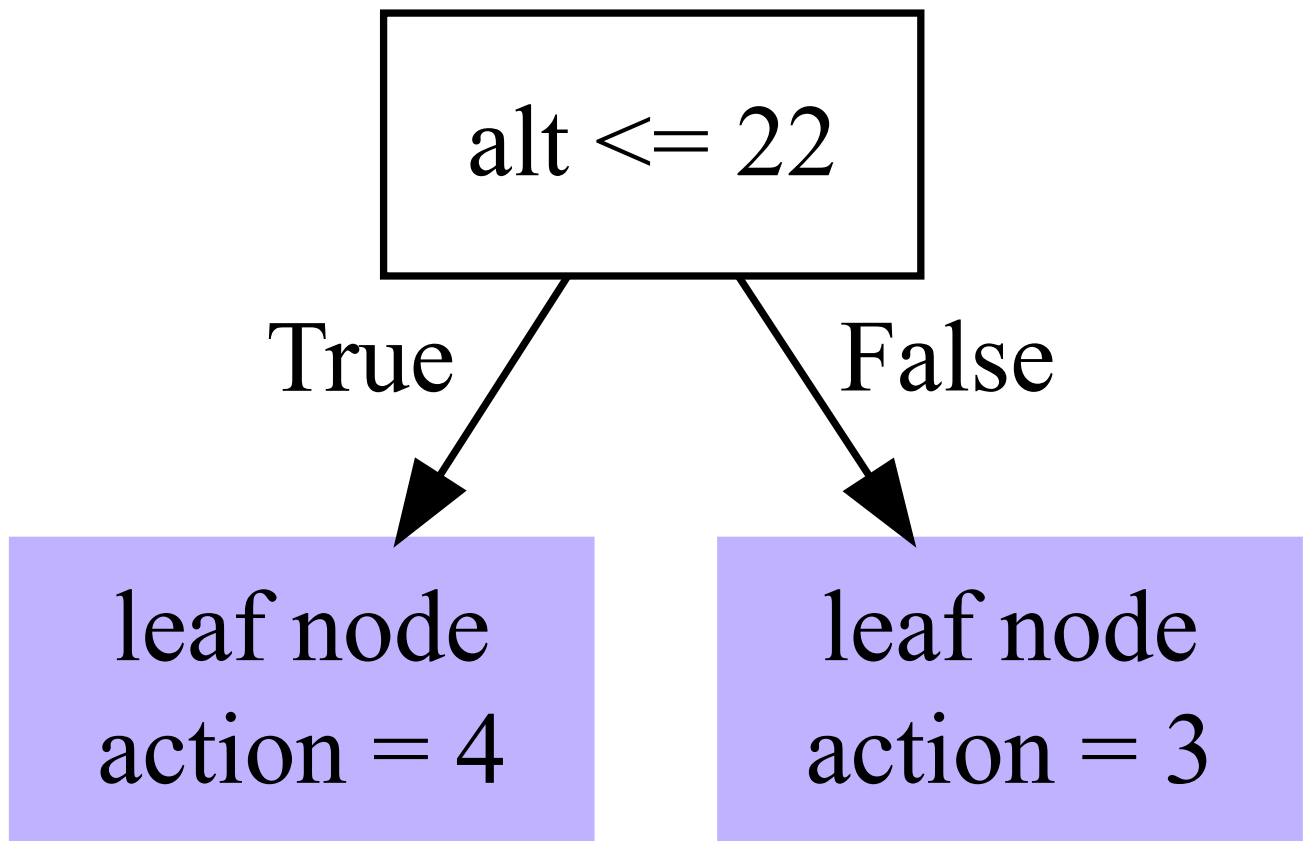
```
# Y before translation
# IPW_true<-read.csv("IPW_true.csv")
# Sum of outcome Y in data
sum = 0
sum_true = 0
for (i in 1:sample_size) {
  sum = sum+(IPW[i,predict(tree_cts, X)[i]])
# sum = sum+(IPW[i,predict(tree_cts, X)[i]])
# sum_true = sum_true+(IPW_true[i,predict(tree_cts, X)[i]])
}
# number of samples under each treatment
num_in_trt <- c(0,0,0,0)
print(sum)
```

```
## [1] 47216.13
```

```
for (j in 1:4) {
  for (i in 1:sample_size) {
    if (predict(tree_cts, X)[i]==j) {
      num_in_trt[j] = num_in_trt[j]+1
    }
  }
}
print(num_in_trt)
```

```
## [1] 0 0 247 611
```

```
plot(tree_cts)
```



...