TSIL 2021

Marjolein Fokkema

Born-againtree approach for predicting treatment outcomes

Load libraries:

```
## Load libraries:
library("foreign")
library("glmertree")
library("partykit")
library("dbarts")
sessionInfo()
```

```
## R version 4.1.0 (2021-05-18)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19042)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=Dutch_Netherlands.1252  LC_CTYPE=Dutch_Netherlands.1252
## [3] LC_MONETARY=Dutch_Netherlands.1252 LC_NUMERIC=C
## [5] LC_TIME=Dutch_Netherlands.1252
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
## [1] dbarts_0.9-19   glmertree_0.2-0 partykit_1.2-13 mvtnorm_1.1-2
## [5] libcoin_1.0-8   lme4_1.1-27.1   Matrix_1.3-4    foreign_0.8-81
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.7        Formula_1.2-4     knitr_1.33        magrittr_2.0.1
##  [5] splines_4.1.0     MASS_7.3-54       lattice_0.20-44   rlang_0.4.11
##  [9] minqa_1.2.4       stringr_1.4.0     tools_4.1.0       parallel_4.1.0
## [13] nlme_3.1-152      xfun_0.24         htmltools_0.5.1.1 survival_3.2-11
## [17] yaml_2.2.1        digest_0.6.27     inum_1.0-4        nloptr_1.2.2.2
## [21] rpart_4.1-15      evaluate_0.14     rmarkdown_2.9     stringi_1.6.2
## [25] compiler_4.1.0    boot_1.3-28
```

Experiment 1: IPDMA ADM with and without Short-Term Psychodynamic Psychotherapy

Load data:

```
## Load data:
IPDMA <- read.spss("3. STPP+ADM vs (BSP+)ADM combined.sav", to.data.frame = TRUE)
levels(IPDMA$Condition) <- c("ADM", "ADM+STPP")
sapply(IPDMA, class)
```

```
##           Study      PatientID      Condition            BSP         Gender
##        "factor"      "numeric"       "factor"       "factor"       "factor"
##             Age        MarStat      Education     EducationB        JobStat
##       "numeric"       "factor"       "factor"       "factor"       "factor"
##        Religion          Epdur        PriorTx        PriorEp          HisHos
##        "factor"       "factor"       "factor"       "factor"       "factor"
##  PDcomorbidity  ADcomorbidity ADcomorbidityB ADcomorbidityC           CGIS
##        "factor"       "factor"       "factor"       "factor"      "numeric"
##             GAF           Zanx         HAMD17     rawHAMDpre       zHAMDpre
##       "numeric"      "numeric"       "factor"      "numeric"      "numeric"
##    rawHAMDpost      zHAMDpost      rawHAMDfu        zHAMDfu
##       "numeric"      "numeric"      "numeric"      "numeric"
```

```
IPDMA <- IPDMA[!(is.na(IPDMA$rawHAMDpost)|is.na(IPDMA$JobStat)), ] # completers only
dim(IPDMA)
```

```
## [1] 376  29
```

Experimental approach:

- Perform 10 repeats of 10-fold CV (on observation level)
- Fit default GLM and GLMM trees and evaluate accuracy
- Fit BART and multilevel BART and evaluate accuracy
- Generate outcomes for both treatment for each observation
- Fit born-again GLM and GLMMM trees to those outcomes. To mitigate effect of increases sample size, give each row a weight of 0.5
- In all predictions from multilevel models, random effects are included

```
nfolds <- 10L
nreps <- 10L
tree_size <- MSE <- data.frame(gt = rep(NA, times = nreps*nfolds))
MSE$bart <- MSE$bart_m <- MSE$surr <- MSE$surr_m <- MSE$gmt <- MSE$gt
tree_size$surr <- tree_size$gmt <- tree_size$surr_m <- tree_size$t
set.seed(42)
for (k in 1:nreps) {
  fold_ids <- sample(rep(1:10, times = ceiling(nrow(IPDMA)/nfolds)),
                     size = nrow(IPDMA), replace = TRUE)
  for (i in 1:nfolds) {

    train_dat <- IPDMA[fold_ids != i, ]
    test_dat <- IPDMA[fold_ids == i, ]

    ## Fit default GLM tree
    gt <- glmtree(rawHAMDpost ~ Condition | rawHAMDpre + Gender + Age + JobStat,
                  data = train_dat)
    gt_preds <- predict(gt, newdata = test_dat)
    MSE$gt[(k-1)*10+i] <- mean((gt_preds - test_dat$rawHAMDpost)^2)
    tree_size$gt[(k-1)*10+i] <- (length(gt)-1)/2
```

```
    ## Fit default GLMM tree
    gmt <- lmertree(rawHAMDpost ~ Condition | Study | rawHAMDpre + Gender +
                      Age + JobStat, data = train_dat)
    gmt_preds <- predict(gmt, newdata = test_dat, re.form = NULL)
    MSE$gmt[(k-1)*10+i] <- mean((gmt_preds - test_dat$rawHAMDpost)^2)
    tree_size$gmt[(k-1)*10+i] <- (length(gmt$tree)-1)/2

    ## Fit BART
    br <- bart2(rawHAMDpost ~ rawHAMDpre + Condition + Gender + Age + JobStat,
                data = train_dat, n.trees = 200, keepTrees = TRUE, verbose = FALSE)
    postp <- predict(br, newdata = test_dat, type = "ppd")
    postm <- apply(postp, 2, median)
    MSE$bart[(k-1)*10+i] <- mean((postm - test_dat$rawHAMDpost)^2)

    ## Prepare surrogate data
    surr_dat <- train_dat[ , -which(names(train_dat) == "Condition")]
    surr_dat <- rbind(surr_dat, surr_dat)
    surr_dat$Condition <- factor(rep(c("ADM", "ADM+STPP"), each = nrow(train_dat)))

    ## Fit multilevel BART
    br_vi <- rbart_vi(rawHAMDpost ~ rawHAMDpre + Condition + Gender + Age + JobStat,
                      data = train_dat, group.by = train_dat$Study,
                      n.trees = 200, keepTrees = TRUE, verbose = FALSE,
                      test = rbind(test_dat, surr_dat),
                      group.by.test = c(test_dat$Study, surr_dat$Study))
    postp_vi <- fitted(br_vi, type = "ppd", sample = "test")[1:nrow(test_dat)]
    MSE$bart_vi[(k-1)*10+i] <- mean((postp_vi - test_dat$rawHAMDpost)^2)

    ## Fit born-again GLMM tree
    surr_dat$rawHAMDpost <- fitted(br_vi, type = "ppd", sample = "test")[-(1:nrow(test_dat))]
    gmt_surr <- lmertree(rawHAMDpost ~ Condition | Study | rawHAMDpre + Gender +
                           Age + JobStat, data = surr_dat,
                         weights = rep(.5, times = nrow(surr_dat)))
    surr_m_preds <- predict(gmt_surr, newdata = test_dat, re.form = NULL)
    MSE$surr_m[(k-1)*10+i] <- mean((surr_m_preds - test_dat$rawHAMDpost)^2)
    tree_size$surr_m[(k-1)*10+i] <- (length(gmt_surr$tree)-1)/2

    ## Fit born-again GLM tree
    postp_surr <- predict(br, newdata = surr_dat, type = "ppd")
    surr_dat$rawHAMDpost <- apply(postp_surr, 2, median)
    gt_surr <- glmtree(rawHAMDpost ~ Condition | rawHAMDpre + Gender + Age + JobStat,
                       data = surr_dat, weights = rep(.5, times = nrow(surr_dat)))
    surr_preds <- predict(gt_surr, newdata = test_dat)
    MSE$surr[(k-1)*10+i] <- mean((surr_preds - test_dat$rawHAMDpost)^2)
    tree_size$surr[(k-1)*10+i] <- (length(gt_surr)-1)/2
  }
}
saveRDS(MSE, "MSE_ipdma.RDS")
saveRDS(tree_size, "treesize_ipdma.RDS")
```

```
MSE <- readRDS("MSE_ipdma.RDS")
tree_size <- readRDS("treesize_ipdma.RDS")

## Benchmark
```

```r
var(IPDMA$rawHAMDpost)
```

```
## [1] 89.77962
```

```r
## Evaluate performance of fixed-effects models
sapply(MSE, mean)[c(1, 4, 6)]
```

```
##       gt     surr     bart
## 60.74533 56.73996 55.94623
```

```r
sapply(MSE, sd)[c(1, 4, 6)]
```

```
##       gt     surr     bart
## 14.20744 12.52976 12.77361
```

```r
t.test(Pair(gt, bart) ~ 1, data = MSE)
```

```
##
##  Paired t-test
##
## data:  Pair(gt, bart)
## t = 7.5769, df = 99, p-value = 1.912e-11
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  3.542329 6.055862
## sample estimates:
## mean of the differences
##                4.799095
```

```r
t.test(Pair(gt, surr) ~ 1, data = MSE)
```

```
##
##  Paired t-test
##
## data:  Pair(gt, surr)
## t = 6.581, df = 99, p-value = 2.244e-09
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  2.797716 5.213024
## sample estimates:
## mean of the differences
##                 4.00537
```

```r
sapply(tree_size[ , c(1, 4)], mean)
```

```
##   gt surr
## 2.91 9.94
```

```r
sapply(tree_size[ , c(1, 4)], sd)
```

```
##        gt      surr
## 0.6046119 1.2618729
```

```r
## Evaluate performance of multilevel models
sapply(MSE, mean)[c(2, 3, 5)]
```

```
##      gmt   surr_m  bart_vi
## 46.11635 43.53680 44.93713
```

```
sapply(MSE, sd)[c(2, 3, 5)]
```

```
##      gmt   surr_m  bart_vi
## 12.16009 11.34613 11.66669
```

```
t.test(Pair(surr_m, bart_vi) ~ 1, data = MSE)
```

```
##
##  Paired t-test
##
## data:  Pair(surr_m, bart_vi)
## t = -5.0704, df = 99, p-value = 1.852e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.9483328 -0.8523364
## sample estimates:
## mean of the differences
##               -1.400335
```

```
t.test(Pair(surr_m, gmt) ~ 1, data = MSE)
```

```
##
##  Paired t-test
##
## data:  Pair(surr_m, gmt)
## t = -7.7679, df = 99, p-value = 7.513e-12
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.238462 -1.920638
## sample estimates:
## mean of the differences
##                -2.57955
```

```
sapply(tree_size[ , c(2, 3)], mean)
```

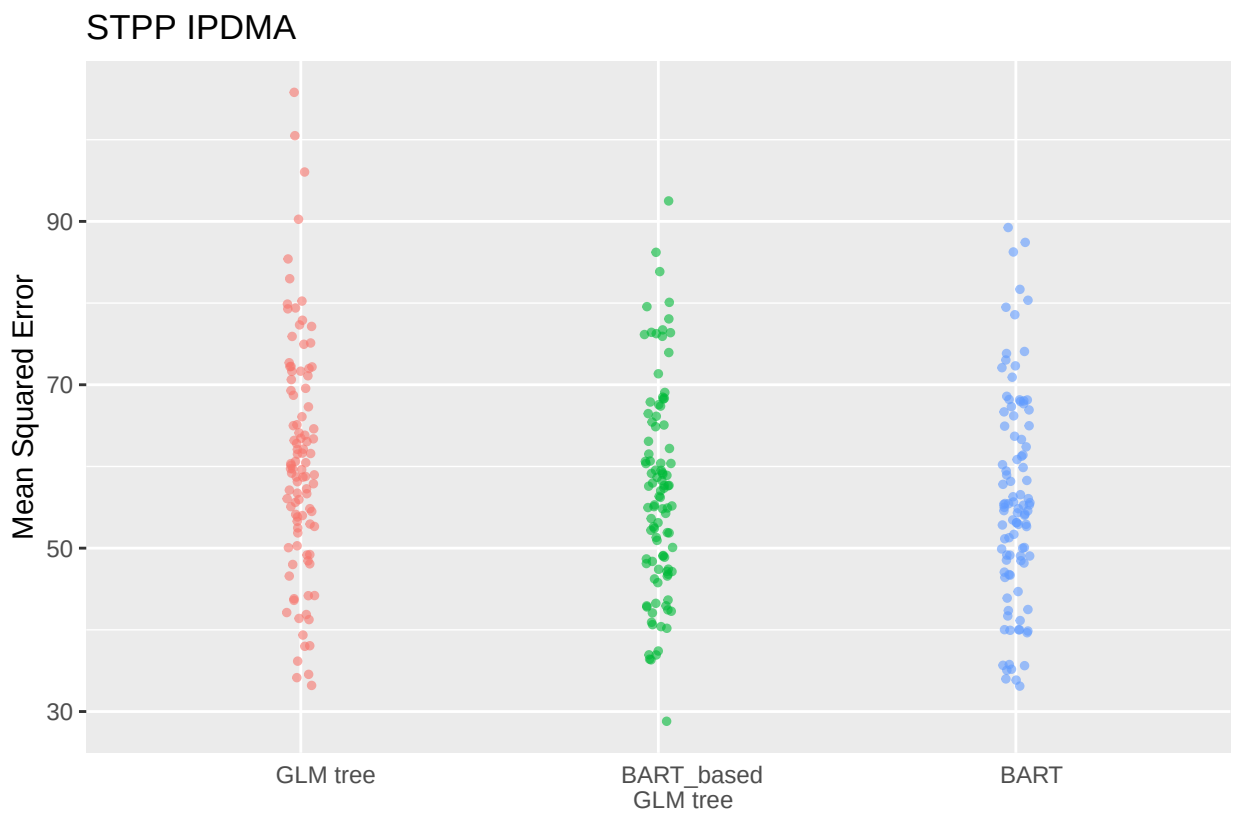
```
##    gmt surr_m
##   2.58   9.53
```

```
sapply(tree_size[ , c(2, 3)], sd)
```

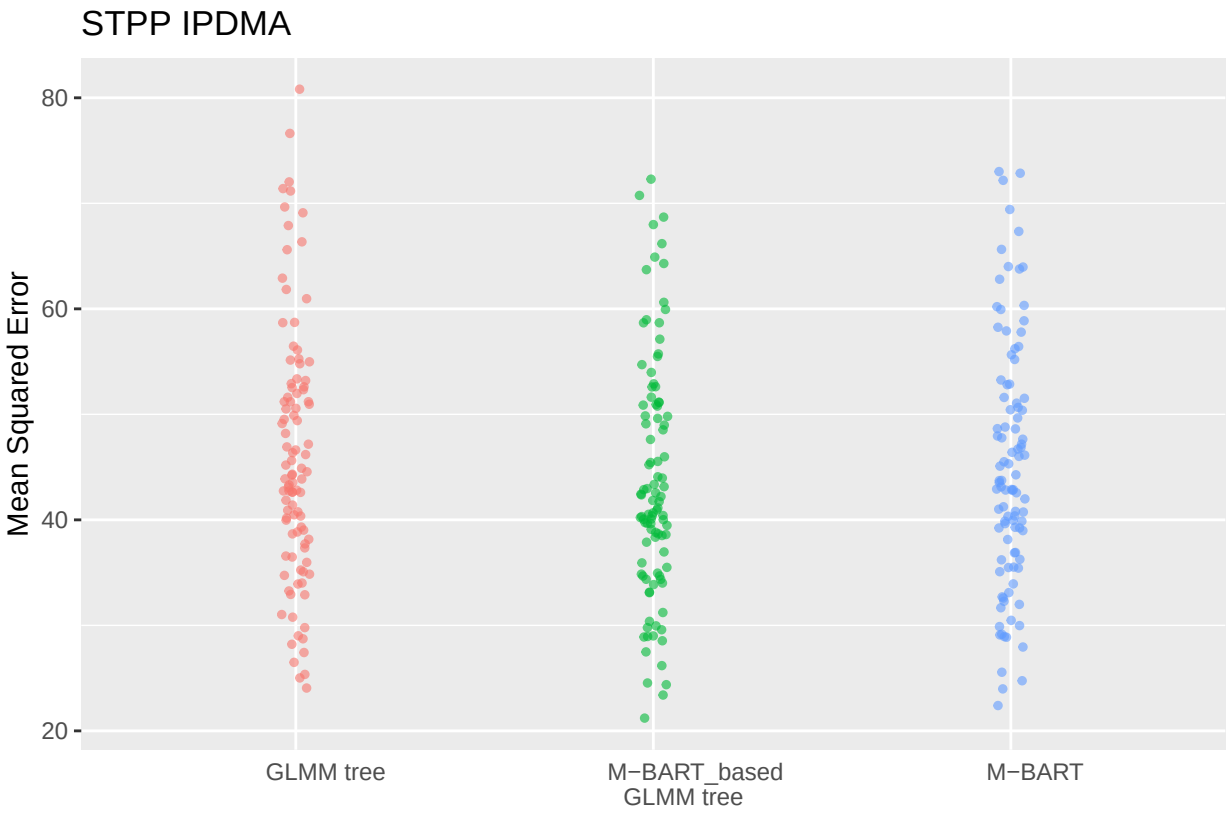
```
##       gmt    surr_m
## 0.5717243 1.2264344
```

Plot performance:

```
library("ggplot2")
MSE_l <- stack(MSE[ , c("gt", "surr", "bart")])
levels(MSE_l$ind) <- c("GLM tree", "BART_based \nGLM tree", "BART")
treesize_l <- stack(tree_size)
ap1 <- ggplot(MSE_l,
              aes(x = ind, y = values, col = ind)) +
  geom_point(position = position_jitterdodge(jitter.width = .2, jitter.height = 0,
                                     dodge.width = 0, seed = 12),
    size = 1, alpha = 0.6) + ylab("Mean Squared Error") + xlab("") +
  ggtitle("STPP IPDMA") +
  theme(axis.text.x = element_text(hjust=.25), axis.ticks.x = element_line(color="white"),
        legend.position = "none")
print(ap1)
```

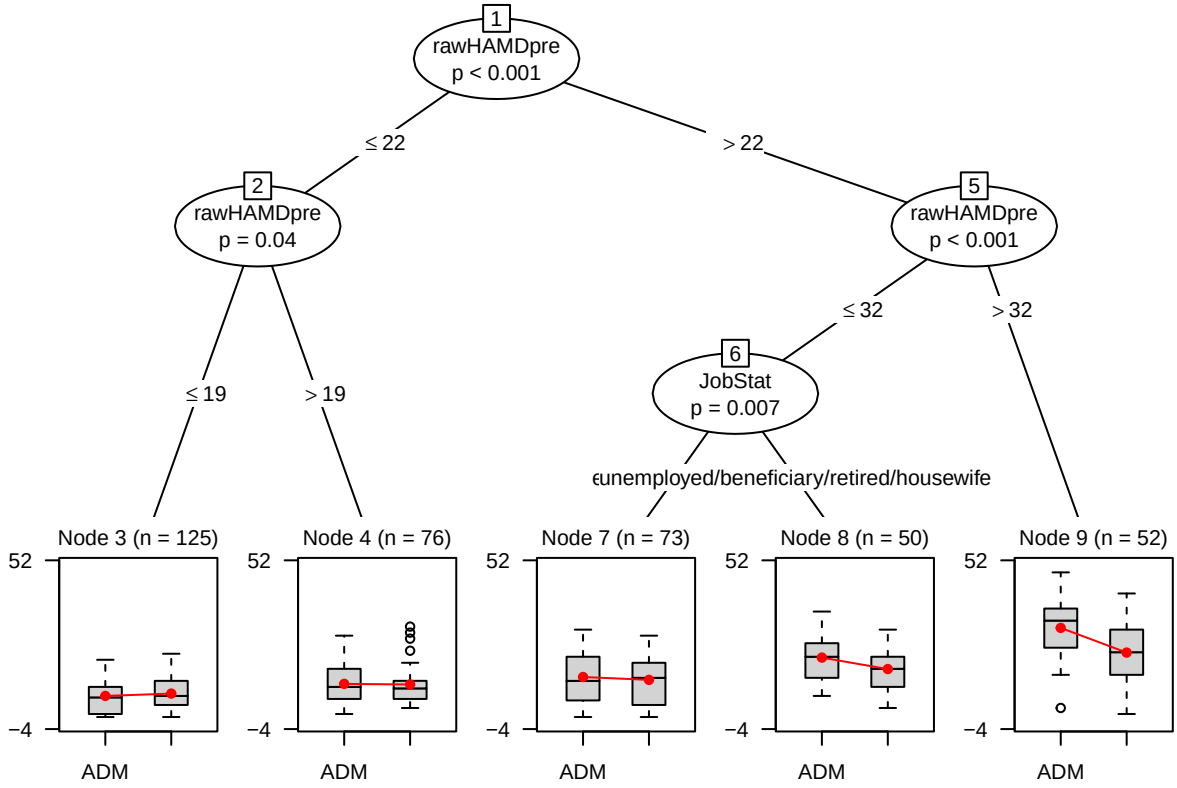



```
MSE_l <- stack(MSE[ , c("gmt", "surr_m", "bart_vi")])
levels(MSE_l$ind) <- c("GLMM tree", "M-BART_based \nGLMM tree", "M-BART")
treesize_l <- stack(tree_size)
ap1 <- ggplot(MSE_l,
              aes(x = ind, y = values, col = ind)) +
  geom_point(position = position_jitterdodge(jitter.width = .2, jitter.height = 0,
                                     dodge.width = 0, seed = 12),
    size = 1, alpha = 0.6) + ylab("Mean Squared Error") + xlab("") +
  ggtitle("STPP IPDMA") +
  theme(axis.text.x = element_text(hjust=.25), axis.ticks.x = element_line(color="white"),
        legend.position = "none")
print(ap1)
```

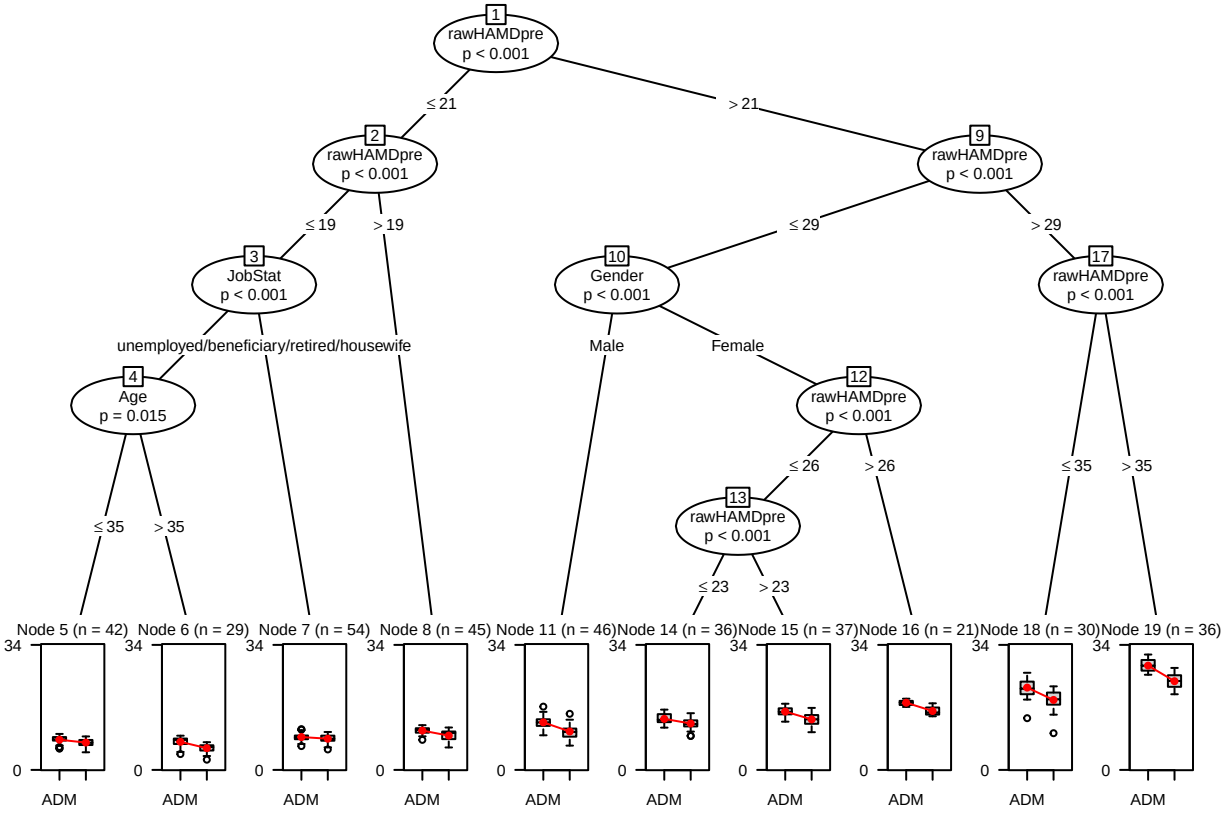


Fit models on complete data:

```
## Fit GLM tree
gt <- glmtree(rawHAMDpost ~ Condition | rawHAMDpre + Gender + Age + JobStat,
              data = IPDMA)
plot(gt, gp = gpar(cex = .7))
```



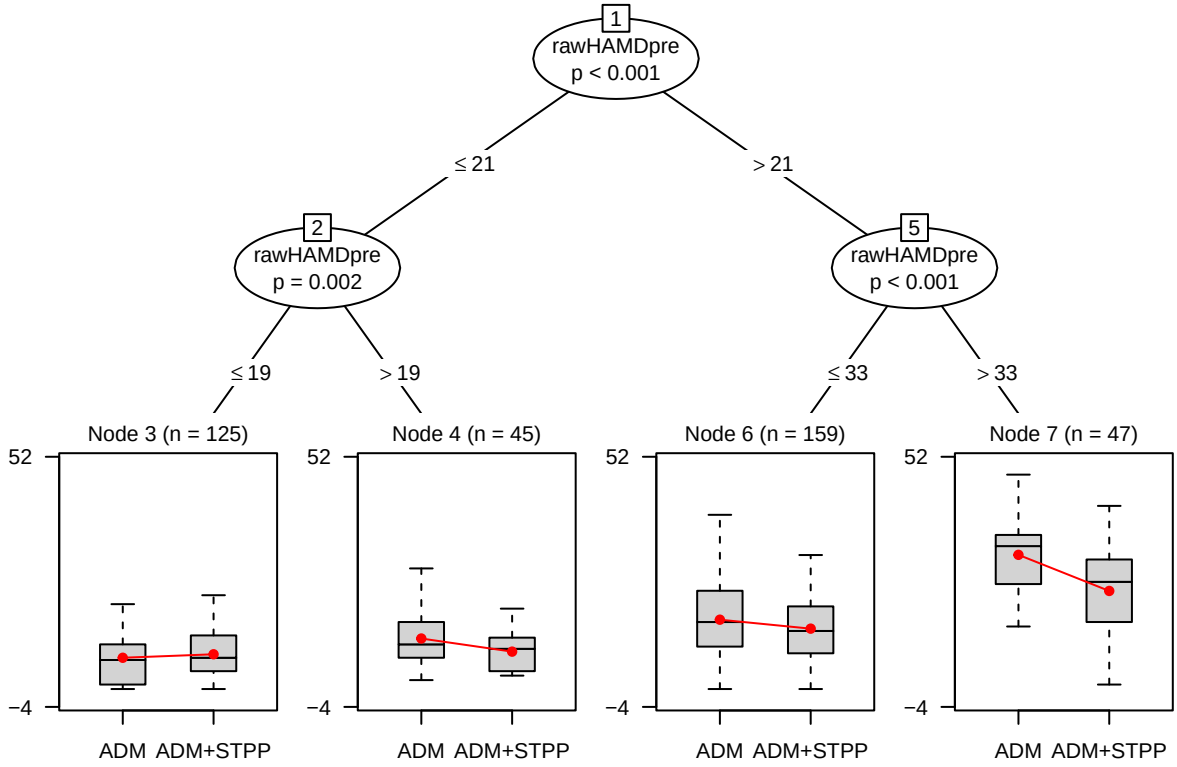

```
## Fit BART
set.seed(42)
br <- bart2(rawHAMDpost ~ Condition +  rawHAMDpre + Gender + Age + JobStat,
            data = IPDMA, n.trees = 200, keepTrees = TRUE, verbose = FALSE)

## Fit surrogate GLM trees
surr_dat <- IPDMA[ , -which(names(IPDMA) == "Condition")]
surr_dat <- rbind(surr_dat, surr_dat)
surr_dat$Condition <- factor(rep(c("ADM", "ADM+STPP"), each = nrow(IPDMA)))
postp_surr <- predict(br, newdata = surr_dat)
surr_dat$rawHAMDpost <- apply(postp_surr, 2, median)
gt_surr <- glmtree(rawHAMDpost ~ Condition | rawHAMDpre + Gender + Age + JobStat,
                   data = surr_dat, weights = rep(.5, times = nrow(surr_dat)))
plot(gt_surr, gp = gpar(cex = .5))
```



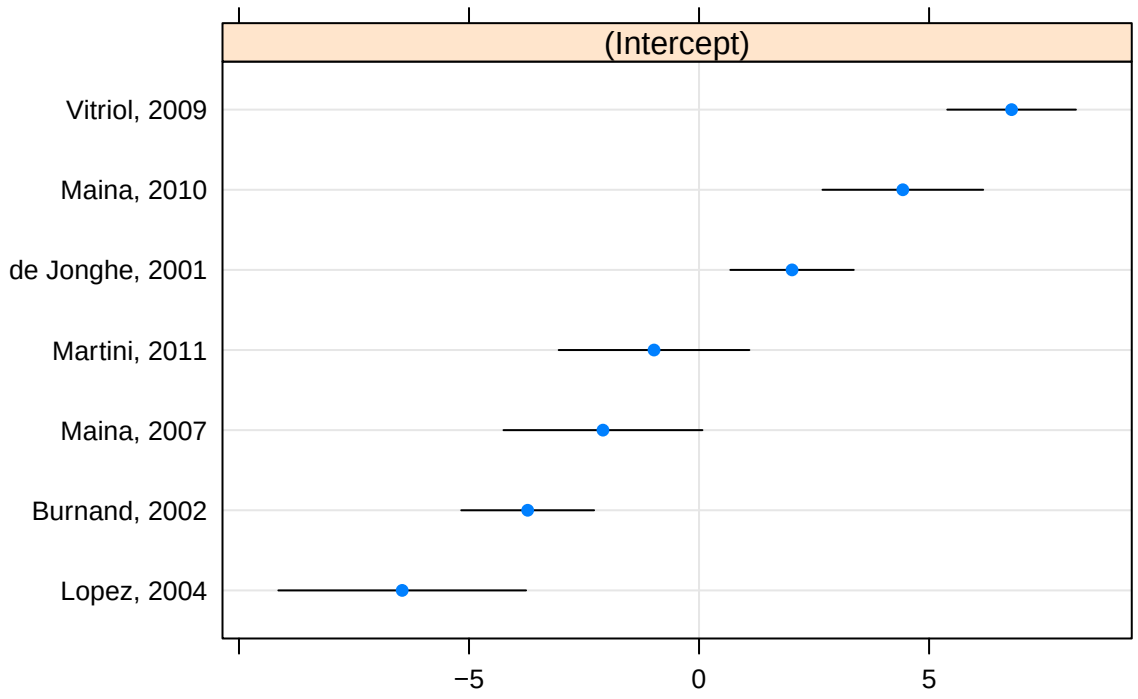

```
## Fit GLMM trees
gt <- lmertree(rawHAMDpost ~ Condition | Study | rawHAMDpre + Gender + Age + JobStat,
               data = IPDMA)
```

```
## Warning in lmertree(rawHAMDpost ~ Condition | Study | rawHAMDpre + Gender + :
## 'data' contains missing values, note that listwise deletion will be employed.
```

```
plot(gt, gp = gpar(cex = .7))
```


1
rawHAMDpre
p < 0.001
≤ 21
> 21
2
rawHAMDpre
p = 0.002
5
rawHAMDpre
p < 0.001
≤ 19
> 19
≤ 33
> 33
Node 3 (n = 125)
Node 4 (n = 45)
Node 6 (n = 159)
Node 7 (n = 47)
52
−4
ADM
ADM+STPP

```
## $Study
```

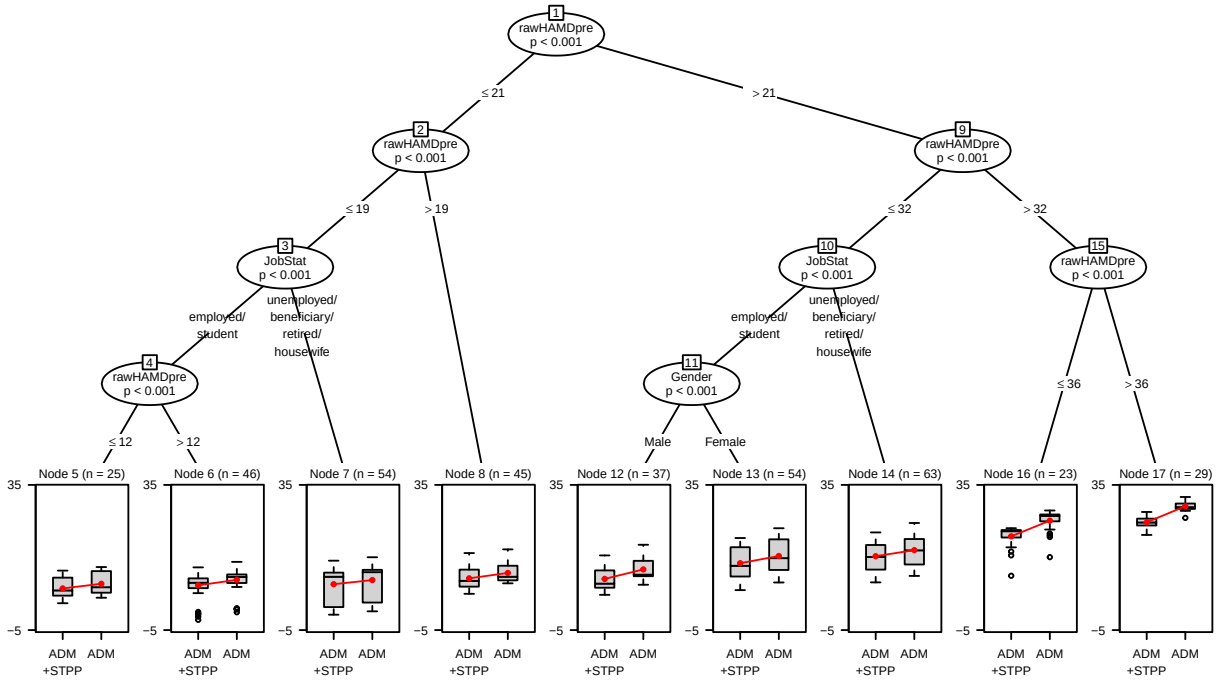



```
## Fit BART VIt
set.seed(42)
br_vi <- rbart_vi(rawHAMDpost ~ rawHAMDpre + Condition + Gender + Age + JobStat,
                  data = IPDMA, group.by = IPDMA$Study, n.trees = 200,
                  keepTrees = TRUE, verbose = FALSE, test = surr_dat,
                  group.by.test = surr_dat$Study)
```

```
## Fit surrogate GLMM tree
surr_dat <- IPDMA[ , -which(names(IPDMA) == "Condition")]
surr_dat <- rbind(surr_dat, surr_dat)
surr_dat$Condition <- factor(rep(c("ADM", "\nADM\n+STPP"), each = nrow(IPDMA)))
surr_dat$rawHAMDpost <- fitted(br_vi, type = "ppd", sample = "test")
levels(surr_dat$JobStat) <- c("employed/\nstudent",
                              "unemployed/\nbeneficiary/\nretired/\nhousewife")
gmt_surr <- lmertree(rawHAMDpost ~ Condition | Study | rawHAMDpre + Gender + Age + JobStat,
                     data = surr_dat, weights = rep(.5, times = nrow(surr_dat)))
```

```
## Warning in lmertree(rawHAMDpost ~ Condition | Study | rawHAMDpre + Gender + :
## 'data' contains missing values, note that listwise deletion will be employed.
```

```
plot(gmt_surr$tree, gp = gpar(cex = .4))
```


1
rawHAMDpre
p < 0.001
≤ 21
> 21
2
rawHAMDpre
p < 0.001
9
rawHAMDpre
p < 0.001
≤ 19
> 19
≤ 32
> 32
3
JobStat
p < 0.001
10
JobStat
p < 0.001
15
rawHAMDpre
p < 0.001
employed/
student
unemployed/
beneficiary/
retired/
housewife
employed/
student
unemployed/
beneficiary/
retired/
housewife
4
rawHAMDpre
p < 0.001
11
Gender
p < 0.001
≤ 36
> 36
≤ 12
> 12
Male
Female
Node 5 (n = 25)
Node 6 (n = 46)
Node 7 (n = 54)
Node 8 (n = 45)
Node 12 (n = 37)
Node 13 (n = 54)
Node 14 (n = 63)
Node 16 (n = 23)
Node 17 (n = 29)
35
−5
ADM
+STPP
ADM

Experiment 2: IPDMA CBT versus PHA

```
## Prepare data
metadata <- read.dta("Database IPDMA CBT PHA Version 11.dta")
metadata[metadata == 999] <- NA
metadata[metadata == 888] <- NA
vars <- c("studyid", "Tx_group", "Age", "Gender", "education",
          "ComorbidAnxietyDisorder", "HRSDt0", "HRSDt1")
factors <- c("studyid", "Tx_group", "Gender", "education", "ComorbidAnxietyDisorder")
metadata$education <- factor(metadata$education, ordered = T)

for (i in 1:length(factors)) {
  metadata[,factors[i]] <- factor(metadata[,factors[i]])
}
metadata <- metadata[vars] # select only relevant variables
metadata <- metadata[complete.cases(metadata[,vars]),] # select only complete data
metadata <- metadata[!metadata$Tx_group == "placebo",] # remove placebo observations
metadata$Tx_group <- factor(metadata$Tx_group)
dim(metadata)
```

```
## [1] 694   8
```

Run the experiment:

```
nreps <- 10L
nfolds <- 10L
set.seed(42)
tree_size <- MSE <- data.frame(gt = rep(NA, times = nreps*nfolds))
MSE$bart <- MSE$bart_vi <- MSE$surr_m <- MSE$surr <- MSE$gmt <- MSE$gt
tree_size$surr <- tree_size$gmt <- tree_size$surr_m <- tree_size$gt
set.seed(42)
for (k in 1:nreps) {
fold_ids <- sample(rep(1:10, times = ceiling(nrow(metadata)/nfolds)),
                                          size = nrow(metadata), replace = TRUE)
  for (i in 1:nfolds) {

    train_dat <- metadata[fold_ids != i, ]
    test_dat <- metadata[fold_ids == i, ]

    ## Fit GLM tree
    gt <- glmtree(HRSDt1 ~ Tx_group | HRSDt0 + Gender + Age + education +
                    ComorbidAnxietyDisorder, data = train_dat)
    gt_preds <- predict(gt, newdata = test_dat)
    MSE$gt[(k-1)*10+i] <- mean((gt_preds - test_dat$HRSDt1)^2)
    tree_size$gt[(k-1)*10+i] <- (length(gt)-1)/2

    ## Fit GLMM trees
    gmt <- lmertree(HRSDt1 ~ Tx_group | studyid | HRSDt0 + Gender + Age +
                      education + ComorbidAnxietyDisorder, data = train_dat)
    gmt_preds <- predict(gmt, newdata = test_dat, re.form = NULL)
    MSE$gmt[(k-1)*10+i] <- mean((gmt_preds - test_dat$HRSDt1)^2)
    tree_size$gmt[(k-1)*10+i] <- (length(gmt$tree)-1)/2

    ## Fit BART
    br <- bart2(HRSDt1 ~ HRSDt0 + Tx_group + Gender + Age + education +
```

```
                  ComorbidAnxietyDisorder, data = train_dat, n.trees = 200,
                keepTrees = TRUE, verbose = FALSE)
    postp <- predict(br, newdata = test_dat, type = "ppd")
    postm <- apply(postp, 2, median)
    MSE$bart[(k-1)*10+i] <- mean((postm - test_dat$HRSDt1)^2)

    ## Prepare surrogate data
    surr_dat <- train_dat[ , -which(names(train_dat) == "Tx_group")]
    surr_dat <- rbind(surr_dat, surr_dat)
    surr_dat$Tx_group <- factor(rep(c("CBT", "PHA"), each = nrow(train_dat)))

    ## Fit multilevel BART
    br_vi <- rbart_vi(HRSDt1 ~ HRSDt0 + Tx_group + Gender + Age + education +
                        ComorbidAnxietyDisorder, data = train_dat,
                      group.by = train_dat$studyid, n.trees = 200, keepTrees = TRUE,
                      verbose = FALSE, test = rbind(test_dat, surr_dat),
                      group.by.test = c(test_dat$studyid, surr_dat$studyid))
    postp_vi <- fitted(br_vi, type = "ppd", sample = "test")[1:nrow(test_dat)]
    MSE$bart_vi[(k-1)*10+i] <- mean((postp_vi - test_dat$HRSDt1)^2)

    ## Fit born-again GLMM tree
    surr_dat$HRSD_t1 <- fitted(br_vi, type = "ppd", sample = "test")[-(1:nrow(test_dat))]
    gmt_surr <- lmertree(HRSDt1 ~ Tx_group | studyid | HRSDt0 + Gender + Age +
                           education + ComorbidAnxietyDisorder, data = surr_dat,
                         weights = rep(.5, times = nrow(surr_dat)))
    surr_m_preds <- predict(gmt_surr, newdata = test_dat, re.form = NULL)
    MSE$surr_m[(k-1)*10+i] <- mean((surr_m_preds - test_dat$HRSDt1)^2)
    tree_size$surr_m[(k-1)*10+i] <- (length(gmt_surr$tree)-1)/2

    ## Fit born-again GLM trees
    postp_surr <- predict(br, newdata = surr_dat)
    surr_dat$HRSD_t1 <- apply(postp_surr, 2, median)
    gt_surr <- glmtree(HRSDt1 ~ Tx_group | HRSDt0 + Gender + Age + education +
                         ComorbidAnxietyDisorder, data = surr_dat,
                       weights = rep(.5, times = nrow(surr_dat)))
    surr_preds <- predict(gt_surr, newdata = test_dat)
    MSE$surr[(k-1)*10+i] <- mean((surr_preds - test_dat$HRSDt1)^2)
    tree_size$surr[(k-1)*10+i] <- (length(gt_surr)-1)/2
  }
}
saveRDS(MSE, "MSE_metadata.RDS")
saveRDS(tree_size, "treesize_metadata.RDS")
```

```
MSE <- readRDS("MSE_metadata.RDS")
tree_Size <- readRDS("treesize_metadata.RDS")

## Benchmark
var(metadata$HRSDt1)
```

```
## [1] 39.24745
```

```
## Evaluate performance of fixed-effects models
sapply(MSE, mean)[c(1, 3, 6)]
```

```
##       gt     surr     bart
```

```
## 38.13097 37.89664 36.98396
```

```
sapply(MSE, sd)[c(1, 3, 6)]
```

```
##       gt     surr     bart
## 6.168936 6.173103 5.930159
```

```
t.test(Pair(gt, bart) ~ 1, data = MSE)
```

```
##
##  Paired t-test
##
## data:  Pair(gt, bart)
## t = 4.7578, df = 99, p-value = 6.672e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.6686546 1.6253563
## sample estimates:
## mean of the differences
##                1.147005
```

```
t.test(Pair(gt, surr) ~ 1, data = MSE)
```

```
##
##  Paired t-test
##
## data:  Pair(gt, surr)
## t = 1.1932, df = 99, p-value = 0.2356
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1553488  0.6239997
## sample estimates:
## mean of the differences
##               0.2343254
```

```
sapply(tree_size[ , c(1, 4)], mean)
```

```
##   gt surr
## 2.91 9.94
```

```
sapply(tree_size[ , c(1, 4)], sd)
```

```
##        gt      surr
## 0.6046119 1.2618729
```

```
## Evaluate performance of multilevel models
sapply(MSE, mean)[c(2, 4, 5)]
```

```
##      gmt   surr_m  bart_vi
## 36.38229 35.91066 35.84005
```

```
sapply(MSE, sd)[c(2, 4, 5)]
```

```
##      gmt   surr_m  bart_vi
## 5.857651 5.752931 5.658027
```

```
t.test(Pair(surr_m, bart_vi) ~ 1, data = MSE)
```

```
##
##  Paired t-test
```

```
##
## data:  Pair(surr_m, bart_vi)
## t = 0.31784, df = 99, p-value = 0.7513
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.3701942  0.5114130
## sample estimates:
## mean of the differences
##              0.07060941
```

```
t.test(Pair(surr_m, gmt) ~ 1, data = MSE)
```

```
##
##  Paired t-test
##
## data:  Pair(surr_m, gmt)
## t = -2.415, df = 99, p-value = 0.01757
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.85913135 -0.08412538
## sample estimates:
## mean of the differences
##              -0.4716284
```

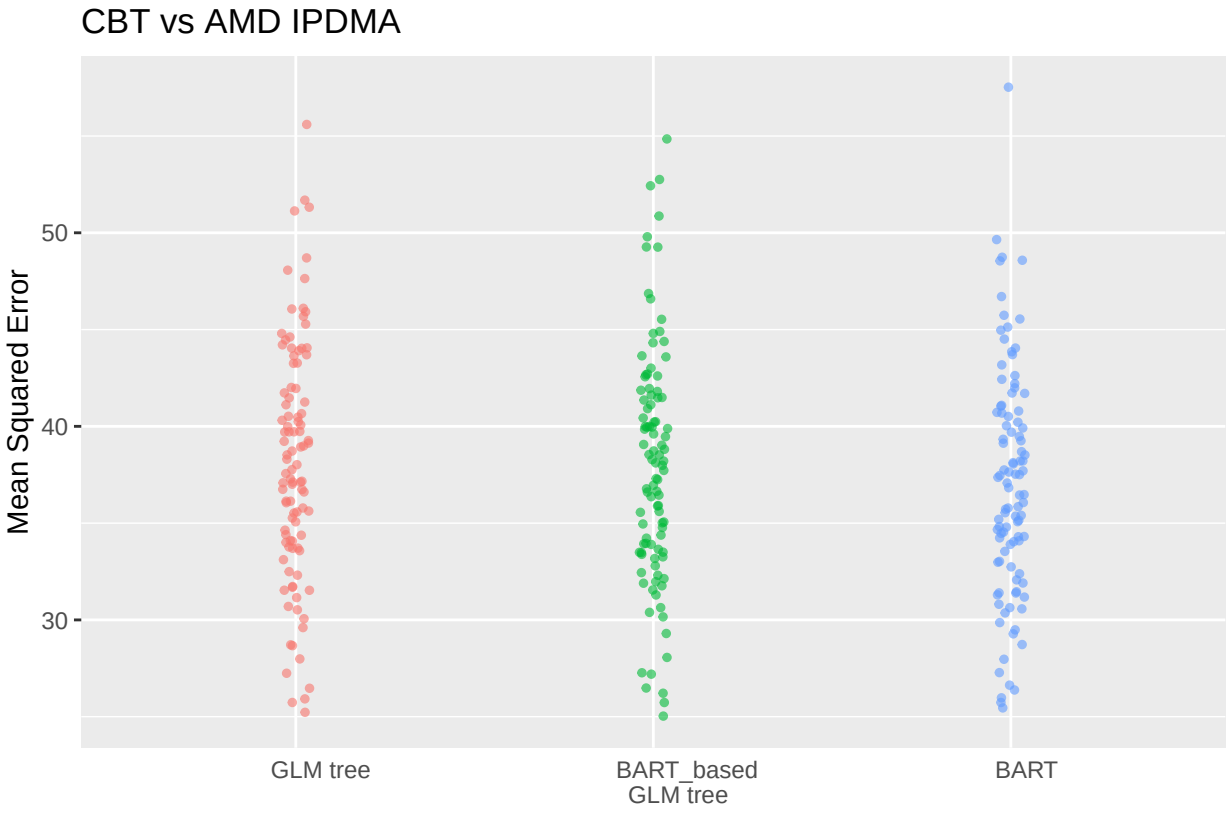
```
sapply(tree_size[ , c(2, 3)], mean)
```

```
##    gmt surr_m
##   2.58   9.53
```

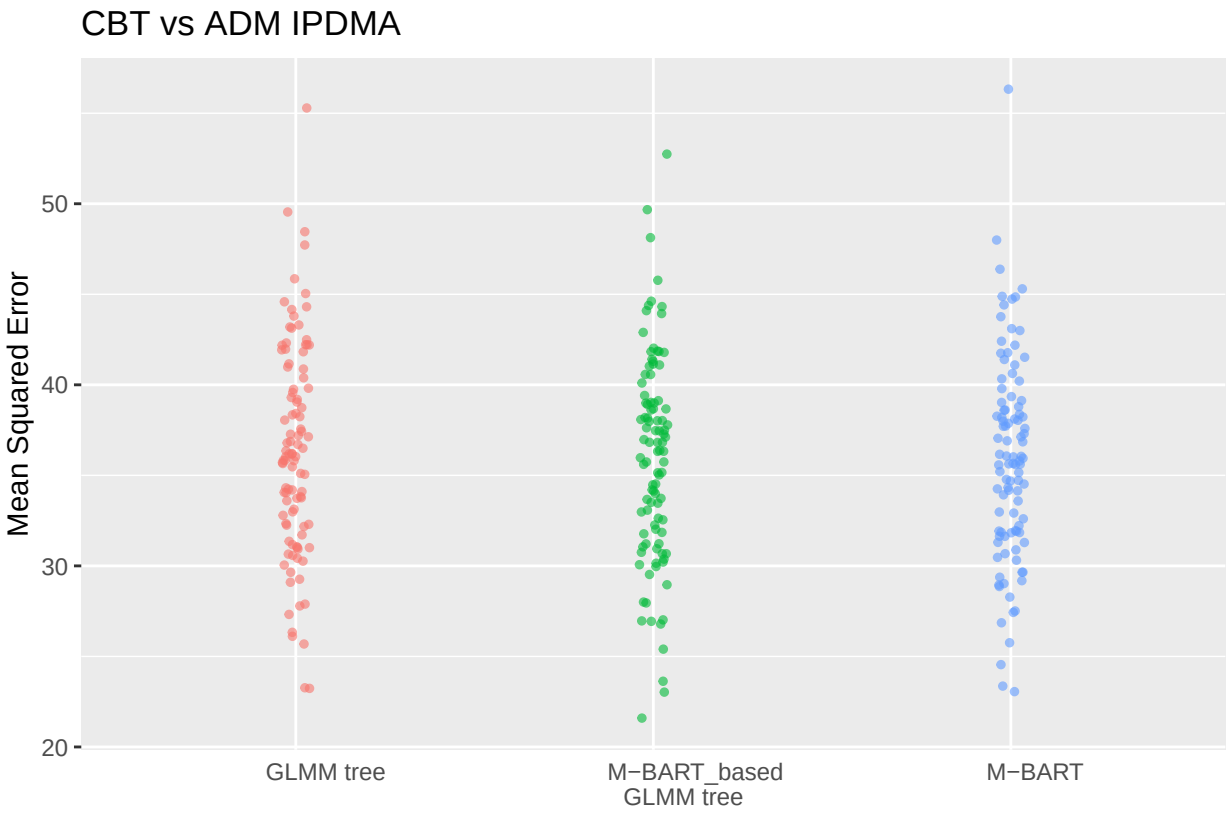
```
sapply(tree_size[ , c(2, 3)], sd)
```

```
##       gmt    surr_m
## 0.5717243 1.2264344
```

```
MSE_l <- stack(MSE[ , c("gt", "surr", "bart")])
levels(MSE_l$ind) <- c("GLM tree", "BART_based \nGLM tree", "BART")
treesize_l <- stack(tree_size)
ap1 <- ggplot(MSE_l,
              aes(x = ind, y = values, col = ind)) +
  geom_point(position = position_jitterdodge(jitter.width = .2, jitter.height = 0,
                                     dodge.width = 0, seed = 12),
    size = 1, alpha = 0.6) + ylab("Mean Squared Error") + xlab("") +
  ggtitle("CBT vs AMD IPDMA") +
  theme(axis.text.x = element_text(hjust=.25), axis.ticks.x = element_line(color="white"),
        legend.position = "none")
print(ap1)
```

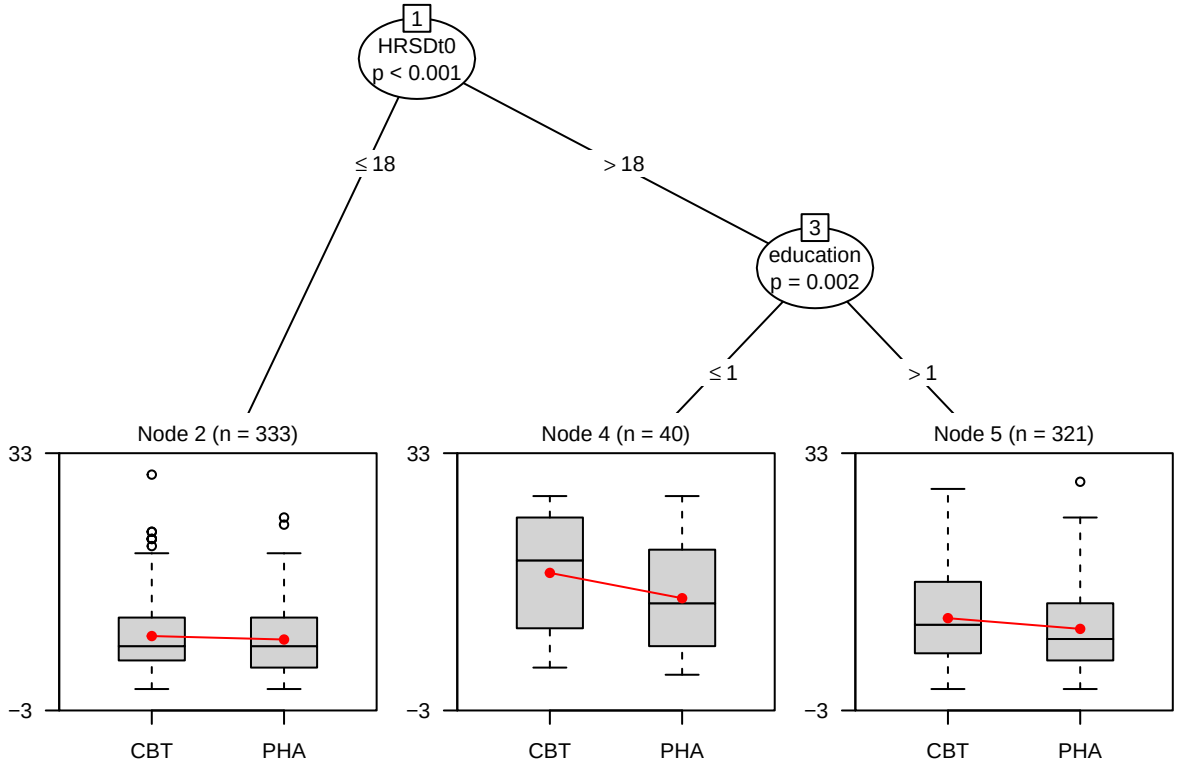



```
MSE_l <- stack(MSE[ , c("gmt", "surr_m", "bart_vi")])
levels(MSE_l$ind) <- c("GLMM tree", "M-BART_based \nGLMM tree", "M-BART")
treesize_l <- stack(tree_size)
ap1 <- ggplot(MSE_l,
              aes(x = ind, y = values, col = ind)) +
  geom_point(position = position_jitterdodge(jitter.width = .2, jitter.height = 0,
                                     dodge.width = 0, seed = 12),
    size = 1, alpha = 0.6) + ylab("Mean Squared Error") + xlab("") +
  ggtitle("CBT vs ADM IPDMA") +
  theme(axis.text.x = element_text(hjust=.25), axis.ticks.x = element_line(color="white"),
        legend.position = "none")
print(ap1)
```

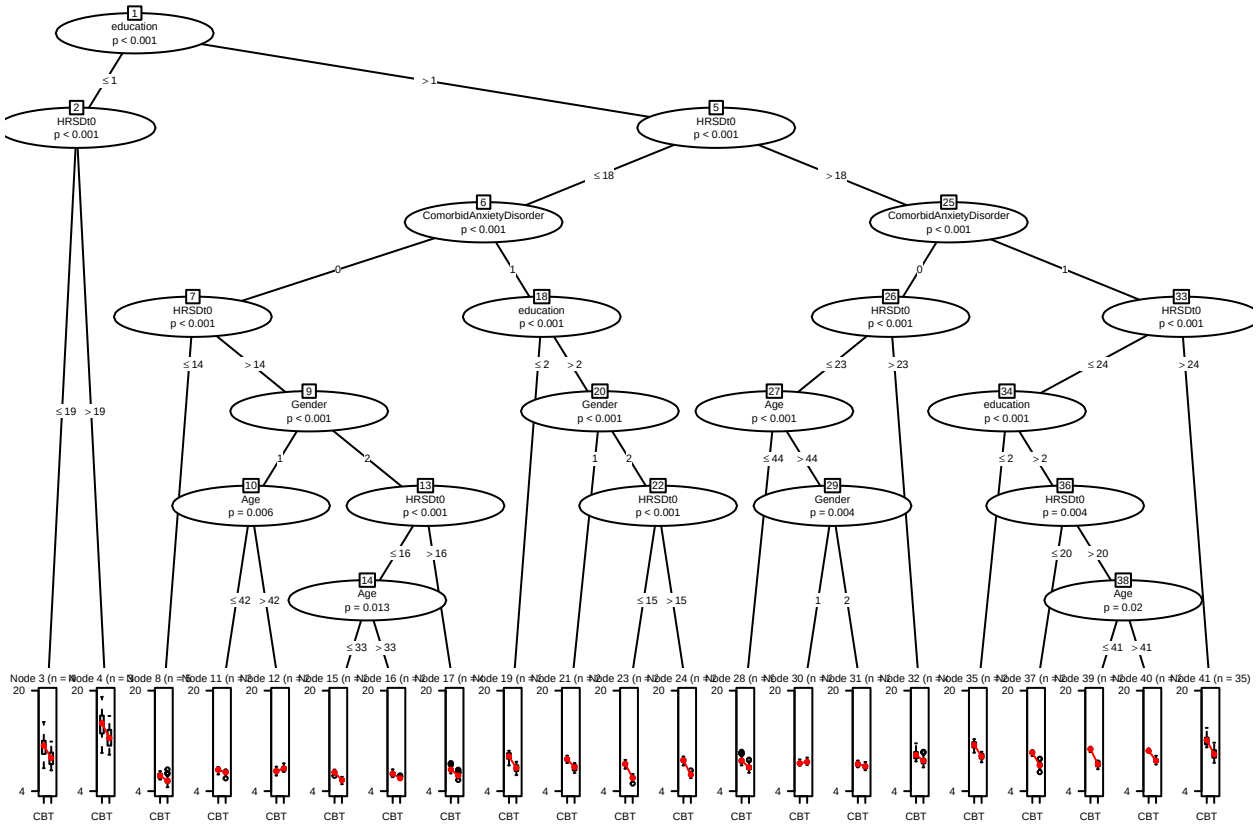


Fit models on complete data:

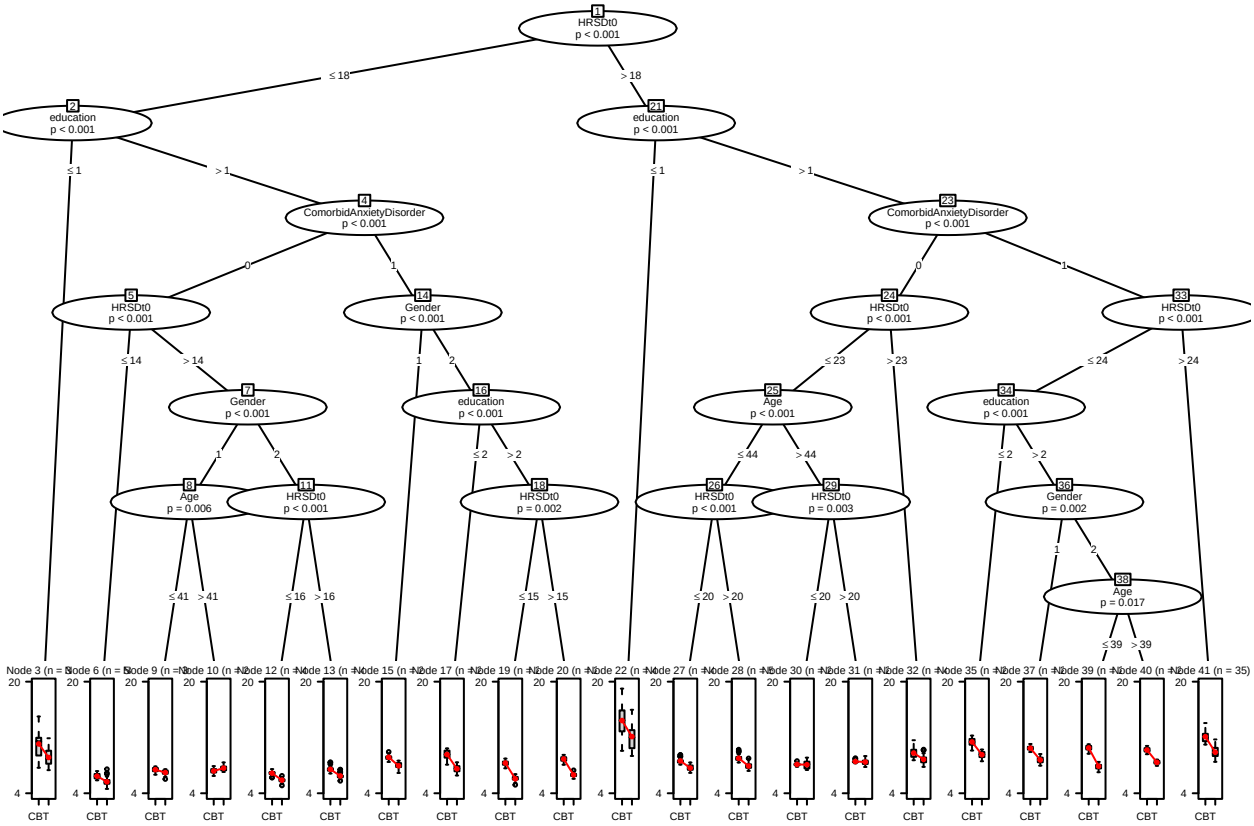
```
## Fit GLM tree
gt <- glmtree(HRSDt1 ~ Tx_group | HRSDt0 + Gender + Age + education +
                ComorbidAnxietyDisorder, data = metadata)
plot(gt, gp = gpar(cex = .7))
```




```
## Fit BART
set.seed(42)
br <- bart2(HRSDt1 ~ Tx_group + HRSDt0 + Gender + Age + education +
            ComorbidAnxietyDisorder, data = metadata, n.trees = 200,
            keepTrees = TRUE, verbose = FALSE)

## Fit surrogate GLM tree
surr_dat <- metadata[ , -which(names(metadata) == "Tx_group")]
surr_dat <- rbind(surr_dat, surr_dat)
surr_dat$Tx_group <- factor(rep(c("CBT", "PHA"), each = nrow(metadata)))
postp_surr <- predict(br, newdata = surr_dat)
surr_dat$HRSDt1 <- apply(postp_surr, 2, median)
gt_surr <- glmtree(HRSDt1 ~ Tx_group | HRSDt0 + Gender + Age + education +
                     ComorbidAnxietyDisorder, data = surr_dat,
                   weights = rep(.5, times = nrow(surr_dat)))
plot(gt_surr, gp = gpar(cex = .35))
```




```
## Fit born-again GLMM tree
set.seed(42)
br_vi <- rbart_vi(HRSDt1 ~ HRSDt0 + Tx_group + Gender + Age + education +
                  ComorbidAnxietyDisorder, data = metadata,
                  group.by = metadata$studyid, n.trees = 200, keepTrees = TRUE,
                  verbose = FALSE, test = surr_dat, group.by.test = surr_dat$studyid)
surr_dat$HRSD_t1 <- fitted(br_vi, type = "ppd", sample = "test")
gmt_surr <- lmertree(HRSDt1 ~ Tx_group | studyid | HRSDt0 + Gender + Age + education +
                     ComorbidAnxietyDisorder, data = surr_dat,
                     weights = rep(.5, times = nrow(surr_dat)))
plot(gmt_surr$tree, gp = gpar(cex=.3))
```


1
HRSDt0
p < 0.001
≤ 18
> 18
2
education
p < 0.001
21
education
p < 0.001
≤ 1
> 1
4
ComorbidAnxietyDisorder
p < 0.001
23
ComorbidAnxietyDisorder
p < 0.001
0
1
5
HRSDt0
p < 0.001
14
Gender
p < 0.001
24
HRSDt0
p < 0.001
33
HRSDt0
p < 0.001
≤ 14
> 14
1
2
≤ 23
> 23
≤ 24
> 24
7
Gender
p < 0.001
16
education
p < 0.001
25
Age
p < 0.001
34
education
p < 0.001
1
2
≤ 2
> 2
≤ 44
> 44
≤ 2
> 2
8
Age
p = 0.006
11
HRSDt0
p < 0.001
18
HRSDt0
p = 0.002
26
HRSDt0
p < 0.001
29
HRSDt0
p = 0.003
36
Gender
p = 0.002
1
2
≤ 41
> 41
≤ 16
> 16
≤ 15
> 15
≤ 20
> 20
≤ 20
> 20
38
Age
p = 0.017
≤ 39
> 39
Node 3 (n = 8)
Node 6 (n = 8)
Node 9 (n = 8)
Node 10 (n = 12)
Node 12 (n = 14)
Node 13 (n = 14)
Node 15 (n = 12)
Node 17 (n = 12)
Node 19 (n = 12)
Node 20 (n = 12)
Node 22 (n = 14)
Node 27 (n = 14)
Node 28 (n = 12)
Node 30 (n = 12)
Node 31 (n = 12)
Node 32 (n = 12)
Node 35 (n = 12)
Node 37 (n = 12)
Node 39 (n = 12)
Node 40 (n = 12)
Node 41 (n = 35)
20
4
CBT