Bart-Based Born-Again Trees

David Eleveld s1741225

18-4-2022

Load requiered packages

Load the Bayesian data generation function

```
gendata_bay <- function(data, PPD, n_gen){

  if(nrow(data) != ncol(PPD)){stop("Amount of observations in PPD is different from data")}

  index_gen <- sample.int(n = nrow(data), size = n_gen, replace = TRUE)
  x_gen <- data[index_gen,]
  y_gen <- sapply(index_gen, function(i) sample(x = PPD[,i], size = 1))
  # y_gen <- sapply(index_gen, function(i) sample(x = PPD[,i][PPD[,i] >= quantile(PPD[,i], probs = .40)

  return(cbind(x_gen, y_gen))
}
```

Load the Smearing data generation function

```
gendata_smr <- function(data, palt = 0.5, n_gen){

  if(palt < 0 | palt > 1){stop("palt must be bound between between 0 and 1")}

  index_gen <- sample.int(n = nrow(data), size = n_gen, replace = TRUE)
  gen_data <- data[index_gen,]

  if(palt != 0){
    for(i in 1:nrow(gen_data)){

      for(j in 1:ncol(gen_data)){

        if(runif(1) <= palt){gen_data[i,j] <- sample(x = data[,j], size = 1)}
      }
    }
  }
  return(gen_data)
}
```

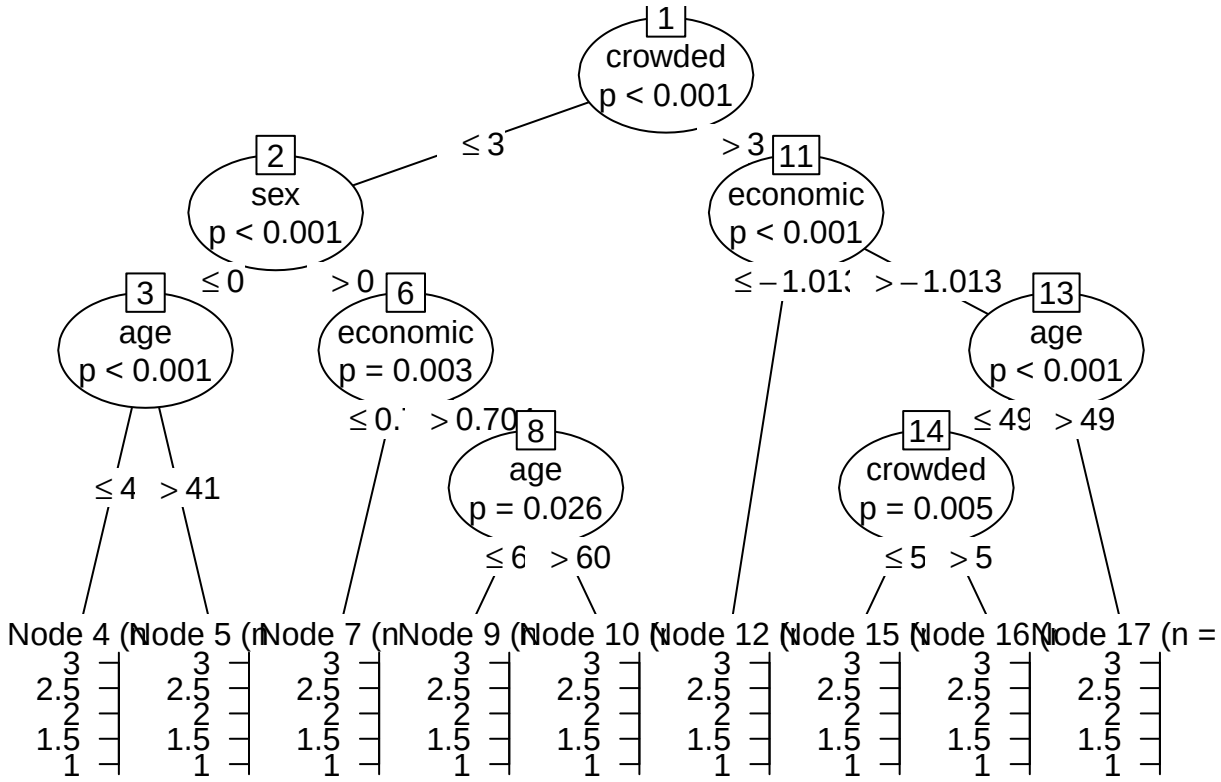
Load Safety data

```
set.seed(42)
data = read_sav(file.path(here(),"Project David","Safety.sav"))
```

```
set.seed(42)
sample <- sample.int(n = nrow(data), size = floor(.75*nrow(data)), replace = F)
traindat <- data[sample, ]
testdat  <- data[-sample, ]
```

Single GLMM tree

```
set.seed(42)
glmtree <- lmertree(unsafe ~ 1 | (1|street) | age + sex + economic + crowded,
                        data = traindat)

pred_glmtree <- predict(glmtree, newdata = testdat)
MSE_glmtree <- mean((pred_glmtree - testdat$unsafe)^2)

plot(glmtree$tree)
```



Multilevel bart model

```
set.seed(42)
Mbart <- rbart_vi(unsafe ~ age + sex + economic + crowded,
         data = traindat, group.by = traindat$street,
         test = testdat, group.by.test = testdat$street,
         n.trees = 100, keepTrees = TRUE)

# This does not work to give the PPD
```

```
PPD_mlbart <- predict(Mbart, newdata = testdat, type = "ppd",
                  group.by = testdat$street, combineChains = TRUE)

# Predicted Values
pred_mlbart <- fitted(object = Mbart, type = "ppd", sample = "test")

MSE_mlbart <- mean((pred_mlbart - testdat$unsafe)^2)
```

Born-again lmertree (Multilevel BART)

```
set.seed(42)
PPD_mlbart_train <- predict(Mbart, newdata = traindat, type = "ppd",
                      group.by = traindat$street, combineChains = TRUE)

surr_mlbart <- gendata_bay(traindat, PPD_mlbart_train, 2 * nrow(traindat))

babart <- lmertree(y_gen ~ 1 | (1|street) | age + sex + economic + crowded,
                       data = surr_mlbart)

pred_babart <- predict(babart, newdata = testdat)

MSE_ba <- mean((pred_babart - testdat$unsafe)^2)
```

Test the model accuracies

```
var(testdat$unsafe)
```

```
## [1] 0.6025542
```

```
MSE_mlbart
```

```
## [1] 0.449172
```

```
MSE_glmtree
```

```
## [1] 0.4844973
```

```
MSE_ba
```

```
## [1] 0.6980035
```

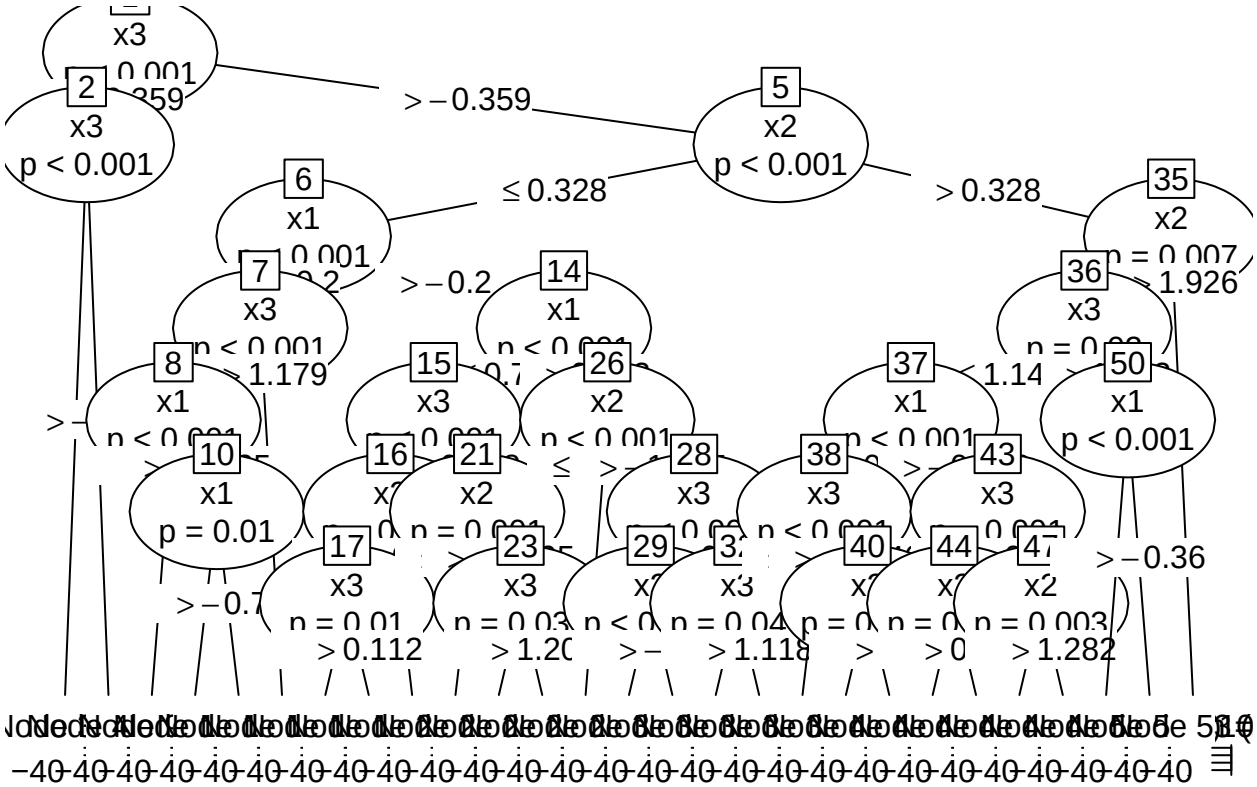
Simulate Data

```
set.seed(42)

N <- 999 #sample size
nG <- 20 #anount of groups

G <- factor(sample(1:nG, N, replace=T)) #grouping variable
```

```r
# level 1 coefs
beta1 <- 1
beta2 <- -2
beta3 <- 3
beta4 <- -4
beta5 <- -1
beta6 <- 2
beta7 <- 3

# level 2 coefs
beta_l2 <- 2

# level 1 vars
x1 <- rnorm(N)
x2 <- rnorm(N)
x3 <- rnorm(N)
err1 <- rnorm(N)

# level 2 vars
tmp <- rnorm(nG)#generate 20 random numbers, m = 0, sd = 1
group <- sapply(G, function(i) tmp[i]) #all units in l2 have the same value
tmp <- rnorm(nG) #error term for level 2
err2 <- sapply(G, function(i) tmp[i]) #all units in l2 have the same value

y <- beta1*x1 + beta2*x2 + beta3*x3 +
  beta4*x1*x2 + beta5*x2*x3 + beta6*x1*x3 + beta7*x1*x2*x3 +
  beta_l2*group +
  err1 + err2

#putting it all together
data <- data.frame(y, group, x1, x2, x3)

sample <- sample.int(n = nrow(data), size = floor(.75*nrow(data)), replace = F)
traindat <- data[sample, ]
testdat  <- data[-sample, ]
```

Single GLMM tree

```r
set.seed(42)
glmtree <- lmertree(y ~ 1 | (1|group) | x1 + x2 + x3,
                        data = traindat)

pred_glmtree <- predict(glmtree, newdata = testdat)
MSE_glmtree <- mean((pred_glmtree - testdat$y)^2)

plot(glmtree$tree)
```



Multilevel bart model

```
set.seed(42)
Mbart <- rbart_vi(y ~ x1 + x2 + x3,
          data = traindat, group.by = traindat$group,
          test = testdat, group.by.test = testdat$group,
          n.trees = 100, keepTrees = TRUE)

# This does not work to give the PPD
PPD_mlbart <- predict(Mbart, newdata = testdat, type = "ppd",
                  group.by = testdat$group, combineChains = TRUE)

# Predicted Values
pred_mlbart <- fitted(object = Mbart, type = "ppd", sample = "test")

MSE_mlbart <- mean((pred_mlbart - testdat$y)^2)
```

Born-again lmertree (Multilevel BART)

```
set.seed(42)
PPD_mlbart_train <- predict(Mbart, newdata = traindat, type = "ppd",
                      group.by = traindat$group, combineChains = TRUE)

surr_mlbart <- gendata_bay(traindat, PPD_mlbart_train, 2 * nrow(traindat))

babart <- lmertree(y_gen ~ 1 | (1|group) | x1 + x2 + x3,
```

```
                        data = surr_mlbart)

pred_babart <- predict(babart, newdata = testdat)

MSE_ba <- mean((pred_babart - testdat$y)^2)
```

Test the model accuracies

```
var(testdat$y)
```

```
## [1] 41.67231
```

```
MSE_mlbart
```

```
## [1] 3.251631
```

```
MSE_glmtree
```

```
## [1] 22.13857
```

```
MSE_ba
```

```
## [1] 118.2236
```