Plots

Marjolein Fokkema

20-7-2022

Boston Housing

```
library("mlbench")
library("ggplot2")
## Compute intercorrelation
data("BostonHousing")
p <- ncol(BostonHousing)-1
sum(cor(sapply(BostonHousing[ , -14L], function(x)
  if (!is.numeric(x)) as.numeric(x) else x))) / (p*(p-1))
```

```
## [1] 0.1266869
```

```
load(file = "BostonHousing MSE.Rda")
load(file = "BostonHousing tree_size.Rda")
#boxplot(MSE)
colMeans(MSE)
```

```
##              GLMM_tree                   Bart                     Ba
##              22.457318               9.478921              21.566889
##            BaBayes_N=1           BaBayes_N=5          BaBayes_N=10
##              23.530386              21.786926              21.623075
##     BaSmear_N=1_palt=0     BaSmear_N=5_palt=0     BaSmear_N=10_palt=0
##              22.484683              21.574473              21.364545
##  BaSmear_N=1_palt=0.25  BaSmear_N=5_palt=0.25  BaSmear_N=10_palt=0.25
##              23.266902              21.820402              21.446998
##   BaSmear_N=1_palt=0.5   BaSmear_N=5_palt=0.5   BaSmear_N=10_palt=0.5
##              26.156004              26.711783              26.284915
##     BaSmear_N=1_palt=1     BaSmear_N=5_palt=1     BaSmear_N=10_palt=1
##              39.813390              42.392009              42.771864
```

```
which.min(colMeans(MSE[ , -2]))
```

```
## BaSmear_N=10_palt=0
##                   8
```

```
sapply(MSE, sd)
```

```
##              GLMM_tree                   Bart                     Ba
##               10.501581               4.938728              11.125934
##            BaBayes_N=1            BaBayes_N=5           BaBayes_N=10
##               10.669522              11.371690              10.048628
##     BaSmear_N=1_palt=0     BaSmear_N=5_palt=0     BaSmear_N=10_palt=0
##                9.738342               9.966675              10.392816
##  BaSmear_N=1_palt=0.25  BaSmear_N=5_palt=0.25  BaSmear_N=10_palt=0.25
##                9.832149               9.026467               8.443231
##   BaSmear_N=1_palt=0.5   BaSmear_N=5_palt=0.5   BaSmear_N=10_palt=0.5
##                9.042027               9.050548               9.187882
##     BaSmear_N=1_palt=1     BaSmear_N=5_palt=1     BaSmear_N=10_palt=1
##               13.099729              12.940042              11.924300
```

```
#boxplot(tree_size)
colMeans(tree_size)
```

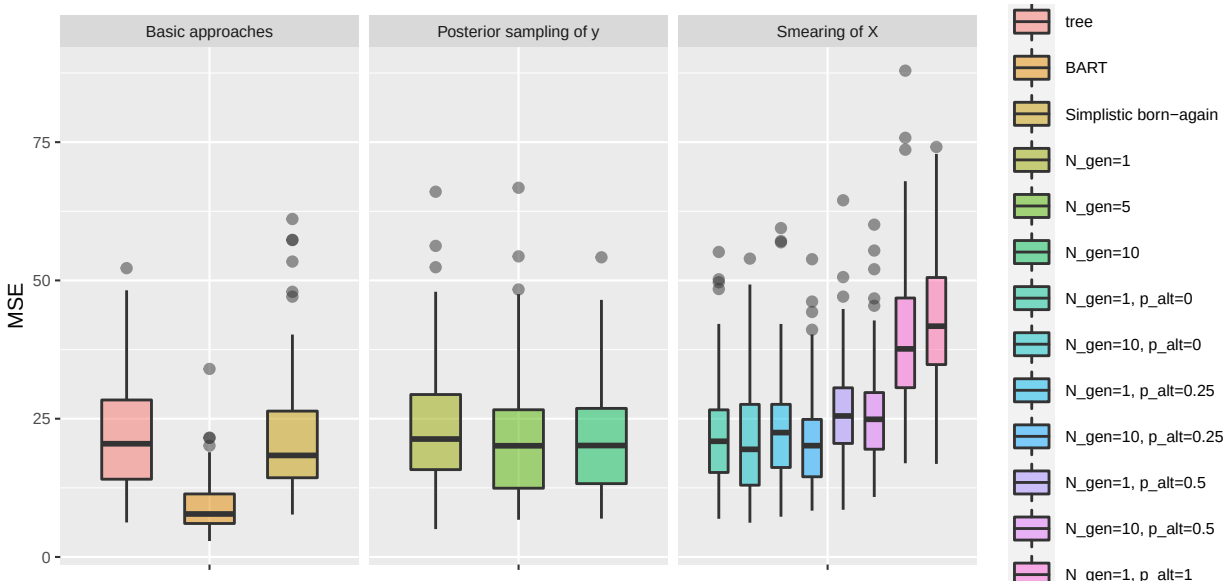
```
##              GLMM_tree                   Bart                     Ba
##                  11.54                     NA                  13.62
##            BaBayes_N=1            BaBayes_N=5           BaBayes_N=10
##                  13.86                  11.57                  11.44
##     BaSmear_N=1_palt=0     BaSmear_N=5_palt=0     BaSmear_N=10_palt=0
##                  17.22                  13.28                  13.45
##  BaSmear_N=1_palt=0.25  BaSmear_N=5_palt=0.25  BaSmear_N=10_palt=0.25
##                  14.06                  10.23                  10.16
##   BaSmear_N=1_palt=0.5   BaSmear_N=5_palt=0.5   BaSmear_N=10_palt=0.5
##                  13.79                   9.86                   9.58
##     BaSmear_N=1_palt=1     BaSmear_N=5_palt=1     BaSmear_N=10_palt=1
##                  13.91                   9.18                   8.79
```

```
sapply(tree_size, sd)
```

```
##              GLMM_tree                   Bart                     Ba
##               1.8280641                     NA              2.1451378
##            BaBayes_N=1            BaBayes_N=5           BaBayes_N=10
##               2.2919270              1.6221634              1.7192904
##     BaSmear_N=1_palt=0     BaSmear_N=5_palt=0     BaSmear_N=10_palt=0
##               2.8161870              2.0893187              1.6537377
##  BaSmear_N=1_palt=0.25  BaSmear_N=5_palt=0.25  BaSmear_N=10_palt=0.25
##               2.0588317              1.2701308              0.9818556
##   BaSmear_N=1_palt=0.5   BaSmear_N=5_palt=0.5   BaSmear_N=10_palt=0.5
##               1.9451831              1.3028330              1.3040729
##     BaSmear_N=1_palt=1     BaSmear_N=5_palt=1     BaSmear_N=10_palt=1
##               1.9389417              1.3210036              0.9670845
```

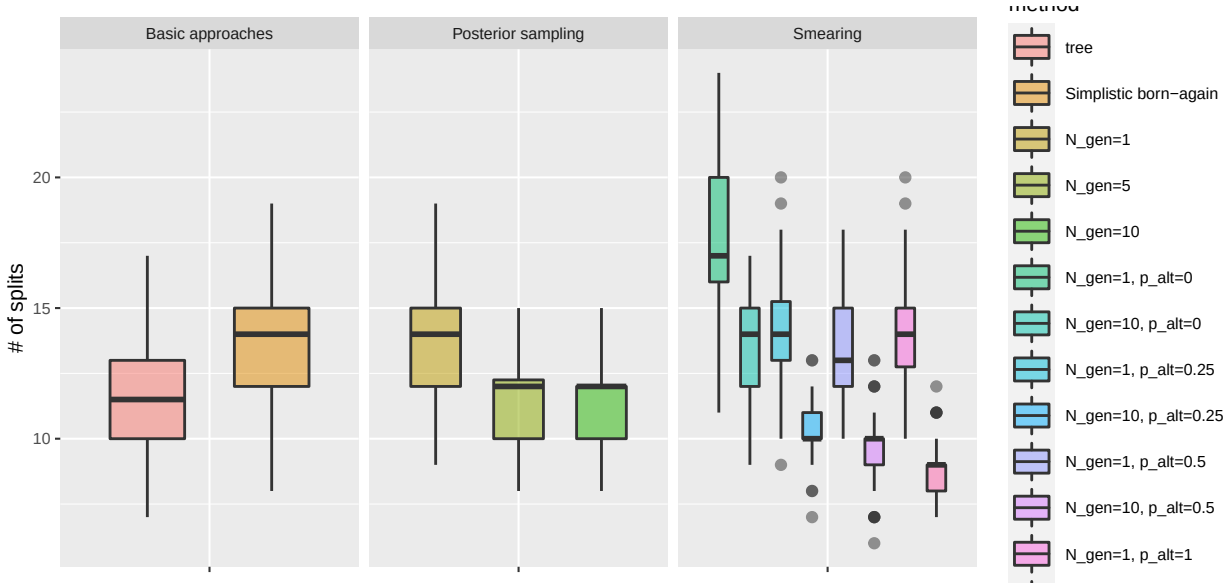
```
theme_set(theme_gray(base_size = 8))

## Create long data and plot
omit_inds <- which(names(MSE) %in% c("BaSmear_N=5_palt=0", "BaSmear_N=5_palt=0.25",
                                     "BaSmear_N=5_palt=0.5", "BaSmear_N=5_palt=1"))

MSE_long <- stack(MSE[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
```

```
names(MSE_long) <- c("MSE", "method")
MSE_long$panel <- ifelse(grepl("BaBayes", MSE_long$method), "Posterior sampling of y",
                         ifelse(grepl("BaSmear", MSE_long$method), "Smearing of X", "Basic approaches
levs <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10",
                           "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                           "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                           "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                           "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
levels(MSE_long$method) <- levs
ggplot(MSE_long) +
  geom_boxplot(aes(x=panel, y=MSE, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "MSE") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```




```
size_long <- stack(tree_size[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(size_long) <- c("size", "method")
size_long$panel <- ifelse(grepl("BaBayes", size_long$method), "Posterior sampling",
                         ifelse(grepl("BaSmear", size_long$method) & grepl("BaSmear", size_long$metho
levels(size_long$method) <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10"
                           "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                           "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                           "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                           "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
ggplot(size_long) +
  geom_boxplot(aes(x=panel, y=size, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "# of splits") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```

```
## Warning: Removed 100 rows containing non-finite values (stat_boxplot).
```



Ozone

```
## Compute intercorrelation
data("Ozone")
p <- ncol(Ozone)-2
sum(cor(sapply(Ozone[ , -c(9L, 13L)], function(x)
  if (!is.numeric(x)) as.numeric(x) else x), use = "pairwise.complete")) / (p*(p-1))
```

```
## [1] 0.1872888
```

```
load(file = "Ozone MSE.Rda")
load(file = "Ozone tree_size.Rda")

sapply(MSE, function(x) table(is.na(x)))
```

```
##       GLMM_tree Bart Ba BaBayes_N=1 BaBayes_N=5 BaBayes_N=10 BaSmear_N=1_palt=0
## FALSE        36   36 36          36          36           36                  36
## TRUE         64   64 64          64          64           64                  64
##       BaSmear_N=5_palt=0 BaSmear_N=10_palt=0 BaSmear_N=1_palt=0.25
## FALSE                 36                  36                    36
## TRUE                  64                  64                    64
##       BaSmear_N=5_palt=0.25 BaSmear_N=10_palt=0.25 BaSmear_N=1_palt=0.5
## FALSE                    36                    36                   36
## TRUE                     64                    64                   64
##       BaSmear_N=5_palt=0.5 BaSmear_N=10_palt=0.5 BaSmear_N=1_palt=1
## FALSE                   36                    36                 36
## TRUE                    64                    64                 64
##       BaSmear_N=5_palt=1 BaSmear_N=10_palt=1
## FALSE                 36                  36
## TRUE                  64                  64
```

```r
#boxplot(MSE)
colMeans(MSE, na.rm=TRUE)
```

```
##              GLMM_tree                   Bart                     Ba
##               22.39551               16.51937               21.09979
##           BaBayes_N=1            BaBayes_N=5           BaBayes_N=10
##               23.55882               21.96742               21.90282
##     BaSmear_N=1_palt=0     BaSmear_N=5_palt=0     BaSmear_N=10_palt=0
##               22.16787               21.15082               20.38417
##  BaSmear_N=1_palt=0.25  BaSmear_N=5_palt=0.25  BaSmear_N=10_palt=0.25
##               23.92898               22.95203               22.92381
##   BaSmear_N=1_palt=0.5   BaSmear_N=5_palt=0.5   BaSmear_N=10_palt=0.5
##               25.95640               25.99136               26.18073
##     BaSmear_N=1_palt=1     BaSmear_N=5_palt=1     BaSmear_N=10_palt=1
##               30.02278               30.43922               30.97491
```

```r
which.min(colMeans(MSE[ , -2]))
```

```
## integer(0)
```

```r
sapply(MSE, sd)
```

```
##              GLMM_tree                   Bart                     Ba
##                     NA                     NA                     NA
##           BaBayes_N=1            BaBayes_N=5           BaBayes_N=10
##                     NA                     NA                     NA
##     BaSmear_N=1_palt=0     BaSmear_N=5_palt=0     BaSmear_N=10_palt=0
##                     NA                     NA                     NA
##  BaSmear_N=1_palt=0.25  BaSmear_N=5_palt=0.25  BaSmear_N=10_palt=0.25
##                     NA                     NA                     NA
##   BaSmear_N=1_palt=0.5   BaSmear_N=5_palt=0.5   BaSmear_N=10_palt=0.5
##                     NA                     NA                     NA
##     BaSmear_N=1_palt=1     BaSmear_N=5_palt=1     BaSmear_N=10_palt=1
##                     NA                     NA                     NA
```

```r
#boxplot(tree_size)
colMeans(tree_size)
```

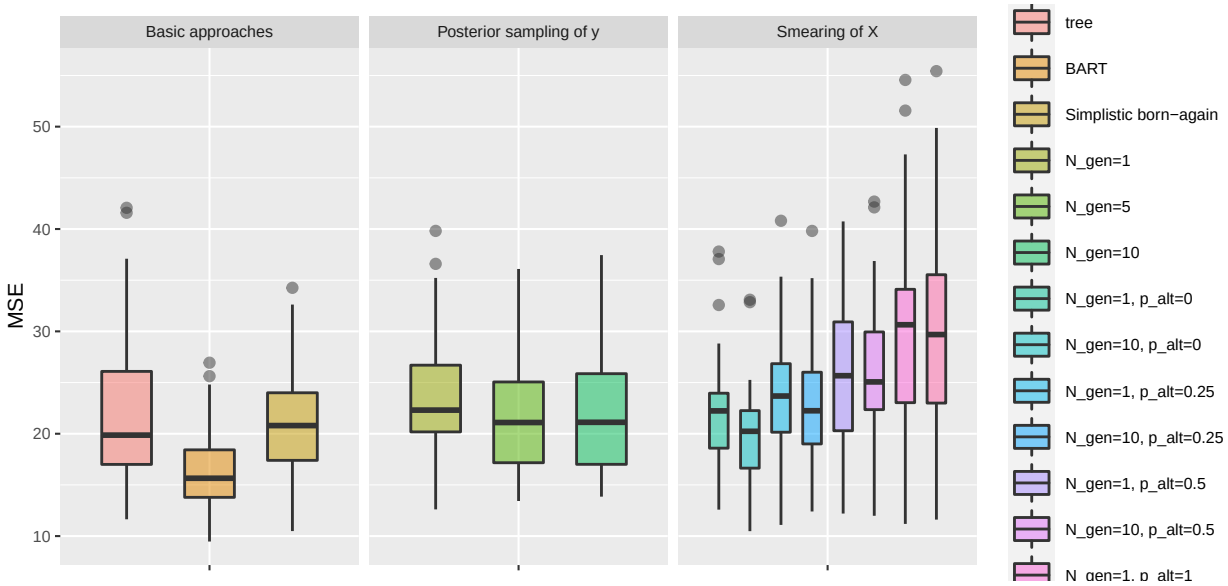
```
##              GLMM_tree                   Bart                     Ba
##                     NA                     NA                     NA
##           BaBayes_N=1            BaBayes_N=5           BaBayes_N=10
##                     NA                     NA                     NA
##     BaSmear_N=1_palt=0     BaSmear_N=5_palt=0     BaSmear_N=10_palt=0
##                     NA                     NA                     NA
##  BaSmear_N=1_palt=0.25  BaSmear_N=5_palt=0.25  BaSmear_N=10_palt=0.25
##                     NA                     NA                     NA
##   BaSmear_N=1_palt=0.5   BaSmear_N=5_palt=0.5   BaSmear_N=10_palt=0.5
##                     NA                     NA                     NA
##     BaSmear_N=1_palt=1     BaSmear_N=5_palt=1     BaSmear_N=10_palt=1
##                     NA                     NA                     NA
```

```
sapply(tree_size, sd)
```

```
##               GLMM_tree                    Bart                      Ba
##                      NA                      NA                      NA
##            BaBayes_N=1            BaBayes_N=5           BaBayes_N=10
##                      NA                      NA                      NA
##     BaSmear_N=1_palt=0     BaSmear_N=5_palt=0    BaSmear_N=10_palt=0
##                      NA                      NA                      NA
##  BaSmear_N=1_palt=0.25  BaSmear_N=5_palt=0.25 BaSmear_N=10_palt=0.25
##                      NA                      NA                      NA
##   BaSmear_N=1_palt=0.5   BaSmear_N=5_palt=0.5  BaSmear_N=10_palt=0.5
##                      NA                      NA                      NA
##     BaSmear_N=1_palt=1     BaSmear_N=5_palt=1    BaSmear_N=10_palt=1
##                      NA                      NA                      NA
```

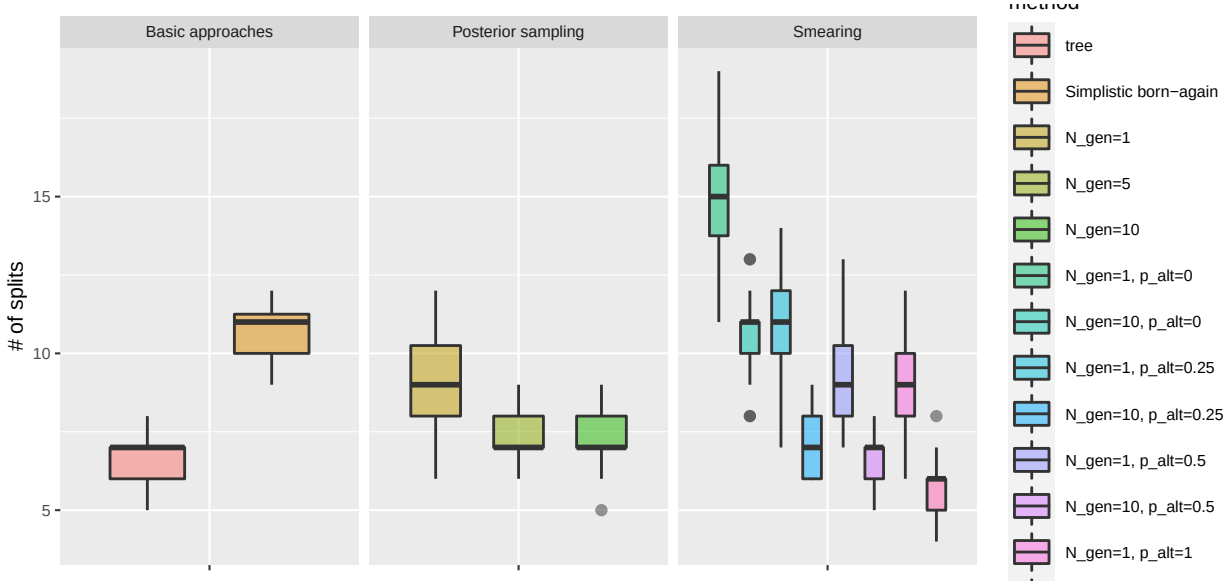
```
theme_set(theme_gray(base_size = 8))

## Create long data and plot
omit_inds <- which(names(MSE) %in% c("BaSmear_N=5_palt=0", "BaSmear_N=5_palt=0.25",
                                     "BaSmear_N=5_palt=0.5", "BaSmear_N=5_palt=1"))

MSE_long <- stack(MSE[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(MSE_long) <- c("MSE", "method")
MSE_long$panel <- ifelse(grepl("BaBayes", MSE_long$method), "Posterior sampling of y",
                         ifelse(grepl("BaSmear", MSE_long$method), "Smearing of X", "Basic approaches

levs <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10",
                              "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                              "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                              "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                              "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
levels(MSE_long$method) <- levs
ggplot(MSE_long) +
  geom_boxplot(aes(x=panel, y=MSE, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "MSE") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```

```
## Warning: Removed 896 rows containing non-finite values (stat_boxplot).
```




```
size_long <- stack(tree_size[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(size_long) <- c("size", "method")
size_long$panel <- ifelse(grepl("BaBayes", size_long$method), "Posterior sampling",
                          ifelse(grepl("BaSmear", size_long$method) & grepl("BaSmear", size_long$metho
levels(size_long$method) <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10"
                              "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                              "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                              "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                              "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
ggplot(size_long) +
  geom_boxplot(aes(x=panel, y=size, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "# of splits") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```

```
## Warning: Removed 932 rows containing non-finite values (stat_boxplot).
```



Friedman

```
library("mlbench")
load(file = "Friedman MSE.Rda")
load(file = "Friedman tree_size.Rda")
set.seed(42)
vars <- c(var(mlbench.friedman1(10000)$y),
          var(mlbench.friedman2(210000)$y),
          var(mlbench.friedman3(10000)$y))
#sapply(MSE, boxplot)
sapply(MSE, colMeans)
```

```
##                             [,1]     [,2]       [,3]
## GLMM_tree              12.076007 33187.34 0.05033159
## Bart                    2.304523 21054.85 0.01880145
## Ba                     11.899889 30487.48 0.04839876
## BaBayes_N=1            13.166062 35566.36 0.05146380
## BaBayes_N=5            12.247483 32711.49 0.05013863
## BaBayes_N=10           12.207703 32719.61 0.05031103
## BaSmear_N=1_palt=0     13.042185 33394.36 0.05041427
## BaSmear_N=5_palt=0     12.073354 31498.65 0.04928297
## BaSmear_N=10_palt=0    11.976251 31029.08 0.04846355
## BaSmear_N=1_palt=0.25  12.075405 31172.79 0.04935003
## BaSmear_N=5_palt=0.25  11.507193 30023.96 0.04778332
## BaSmear_N=10_palt=0.25 11.339046 29879.85 0.04781966
## BaSmear_N=1_palt=0.5   12.034039 31381.05 0.04956370
## BaSmear_N=5_palt=0.5   11.288992 29343.29 0.04745073
## BaSmear_N=10_palt=0.5  11.329361 29413.42 0.04769477
## BaSmear_N=1_palt=1     11.819585 31570.06 0.04928029
## BaSmear_N=5_palt=1     11.198944 29506.09 0.04784267
## BaSmear_N=10_palt=1    11.247026 29137.52 0.04811013
```

```
round((1/vars)*(t(sapply(MSE, colMeans))), digits = 3)
```

```
##      GLMM_tree  Bart    Ba BaBayes_N=1 BaBayes_N=5 BaBayes_N=10
## [1,]     0.486 0.093 0.479       0.530       0.493        0.491
## [2,]     0.208 0.132 0.191       0.223       0.205        0.205
## [3,]     0.442 0.165 0.425       0.451       0.440        0.441
##      BaSmear_N=1_palt=0 BaSmear_N=5_palt=0 BaSmear_N=10_palt=0
## [1,]              0.525              0.486               0.482
## [2,]              0.210              0.198               0.195
## [3,]              0.442              0.432               0.425
##      BaSmear_N=1_palt=0.25 BaSmear_N=5_palt=0.25 BaSmear_N=10_palt=0.25
## [1,]                 0.486                 0.463                  0.456
## [2,]                 0.196                 0.188                  0.187
## [3,]                 0.433                 0.419                  0.420
##      BaSmear_N=1_palt=0.5 BaSmear_N=5_palt=0.5 BaSmear_N=10_palt=0.5
## [1,]                0.484                0.454                 0.456
## [2,]                0.197                0.184                 0.185
## [3,]                0.435                0.416                 0.418
##      BaSmear_N=1_palt=1 BaSmear_N=5_palt=1 BaSmear_N=10_palt=1
## [1,]              0.476              0.451               0.453
## [2,]              0.198              0.185               0.183
## [3,]              0.432              0.420               0.422
```

```
sapply(MSE, function(x) which.min(colMeans(x[ , -2])))
```

```
##   BaSmear_N=5_palt=1  BaSmear_N=10_palt=1 BaSmear_N=5_palt=0.5
##                   16                   17                   13
```

```
sapply(MSE, function(x) sapply(x, sd))
```

```
##                            [,1]     [,2]        [,3]
## GLMM_tree             1.0830838 2599.800 0.006148888
## Bart                  0.2321861 1206.724 0.001907856
## Ba                    1.1238855 2565.068 0.005855916
## BaBayes_N=1           1.5623625 5189.825 0.006404210
## BaBayes_N=5           1.2824767 2559.969 0.005485072
## BaBayes_N=10          1.3374984 3283.079 0.005854500
## BaSmear_N=1_palt=0    1.6714595 3839.703 0.006336858
## BaSmear_N=5_palt=0    1.2009723 2653.883 0.005337941
## BaSmear_N=10_palt=0   1.0352730 2696.107 0.005994306
## BaSmear_N=1_palt=0.25 1.1761710 2496.272 0.005894570
## BaSmear_N=5_palt=0.25 0.8533598 2113.327 0.005512098
## BaSmear_N=10_palt=0.25 0.5798938 2351.263 0.005618629
## BaSmear_N=1_palt=0.5  1.1020613 2541.695 0.005726027
## BaSmear_N=5_palt=0.5  0.7202745 2222.965 0.005854793
## BaSmear_N=10_palt=0.5 0.5822913 2295.298 0.005720670
## BaSmear_N=1_palt=1    1.0387107 2930.827 0.006163060
## BaSmear_N=5_palt=1    0.6842474 2190.938 0.005722531
## BaSmear_N=10_palt=1   0.5267453 1979.481 0.005788078
```

```
#sapply(MSE, boxplot)
sapply(tree_size, colMeans)
```

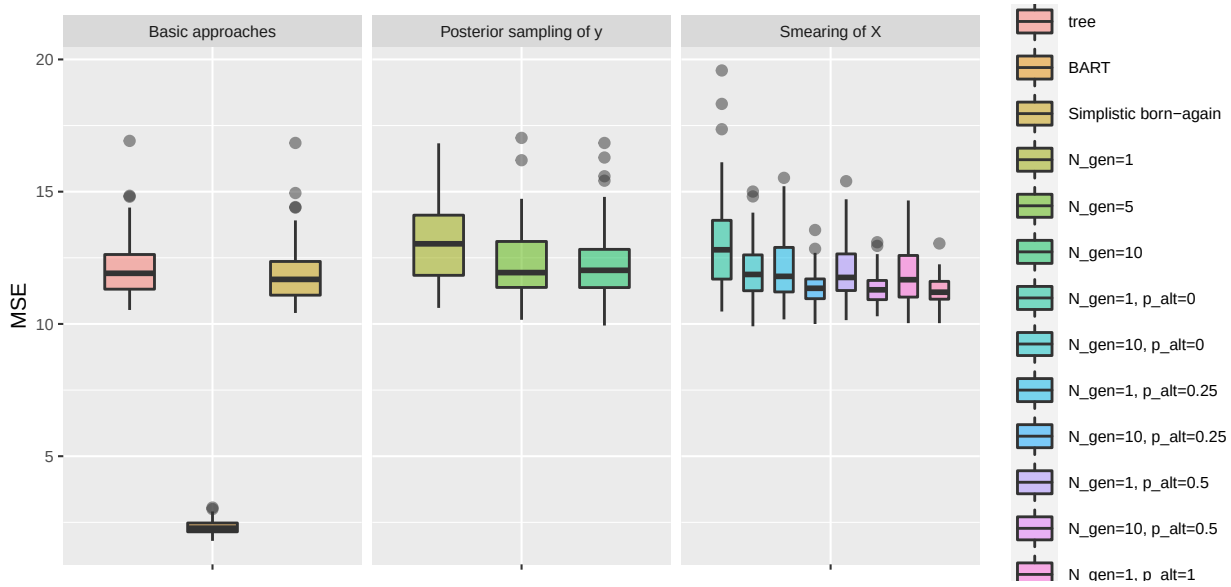
```
##                         [,1]  [,2]  [,3]
## GLMM_tree              6.85  9.02  6.76
## Bart                     NA    NA    NA
## Ba                     7.16 11.88  9.48
## BaBayes_N=1            9.02  9.97  8.08
## BaBayes_N=5            6.95  9.56  7.25
## BaBayes_N=10           6.83  9.61  6.94
## BaSmear_N=1_palt=0     9.01 12.04 10.27
## BaSmear_N=5_palt=0     7.46 11.31  9.33
## BaSmear_N=10_palt=0    7.32 11.48  9.53
## BaSmear_N=1_palt=0.25  7.86 11.95  9.96
## BaSmear_N=5_palt=0.25  5.89 11.67  9.02
## BaSmear_N=10_palt=0.25 5.77 11.32  8.99
## BaSmear_N=1_palt=0.5   7.59 12.02  9.90
## BaSmear_N=5_palt=0.5   5.74 11.85  9.51
## BaSmear_N=10_palt=0.5  5.41 11.83  8.99
## BaSmear_N=1_palt=1     7.63 12.16 10.45
## BaSmear_N=5_palt=1     5.80 11.96  9.47
## BaSmear_N=10_palt=1    5.36 12.13  9.01
```

```
sapply(tree_size, function(x) sapply(x, sd))
```

```
##                            [,1]     [,2]     [,3]
## GLMM_tree              1.1752928 1.189007 1.064771
## Bart                         NA       NA       NA
## Ba                     1.2449494 1.423327 1.359441
## BaBayes_N=1            1.4769611 1.175980 1.186456
## BaBayes_N=5            1.3361712 1.066667 1.057680
## BaBayes_N=10           1.2231356 1.278375 1.052366
## BaSmear_N=1_palt=0     1.5731009 1.392041 1.398809
## BaSmear_N=5_palt=0     1.1670562 1.088925 1.271720
## BaSmear_N=10_palt=0    1.1536503 1.039425 1.266946
## BaSmear_N=1_palt=0.25  1.3028330 1.274260 1.340285
## BaSmear_N=5_palt=0.25  0.8515583 1.082972 1.497338
## BaSmear_N=10_palt=0.25 0.7365631 1.127077 1.290955
## BaSmear_N=1_palt=0.5   1.3111411 1.247057 1.403459
## BaSmear_N=5_palt=0.5   0.7052473 1.209182 1.210184
## BaSmear_N=10_palt=0.5  0.5876679 1.271720 1.329502
## BaSmear_N=1_palt=1     1.3154482 1.276991 1.373450
## BaSmear_N=5_palt=1     0.6816498 1.213726 1.452653
## BaSmear_N=10_palt=1    0.5599423 1.211519 1.424852
```

```
MSE <- MSE[[1]]
tree_size <- tree_size[[1]]



## Create long data and plot
omit_inds <- which(names(MSE) %in% c("BaSmear_N=5_palt=0", "BaSmear_N=5_palt=0.25",
```

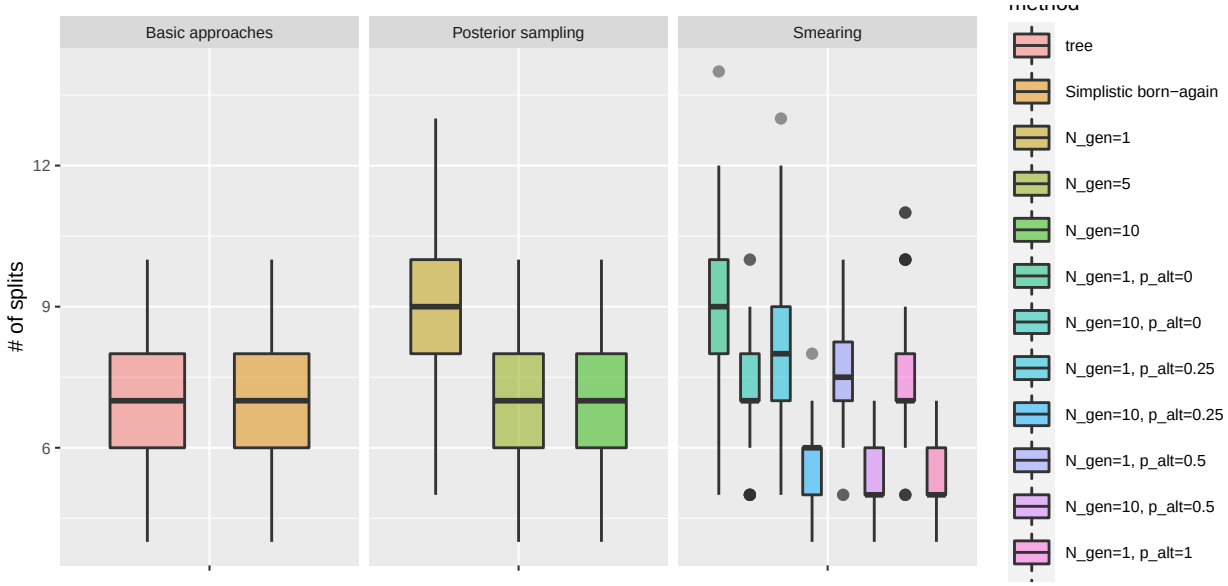
```
                                    "BaSmear_N=5_palt=0.5", "BaSmear_N=5_palt=1"))

MSE_long <- stack(MSE[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(MSE_long) <- c("MSE", "method")
MSE_long$panel <- ifelse(grepl("BaBayes", MSE_long$method), "Posterior sampling of y",
                         ifelse(grepl("BaSmear", MSE_long$method), "Smearing of X", "Basic approaches

levs <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10",
                          "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                          "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                          "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                          "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
levels(MSE_long$method) <- levs
ggplot(MSE_long) +
  geom_boxplot(aes(x=panel, y=MSE, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "MSE") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```




```
size_long <- stack(tree_size[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(size_long) <- c("size", "method")
size_long$panel <- ifelse(grepl("BaBayes", size_long$method), "Posterior sampling",
                         ifelse(grepl("BaSmear", size_long$method) & grepl("BaSmear", size_long$metho
levels(size_long$method) <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10"
                          "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                          "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                          "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                          "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
ggplot(size_long) +
  geom_boxplot(aes(x=panel, y=size, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
```

```
  facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "# of splits") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```

```
## Warning: Removed 100 rows containing non-finite values (stat_boxplot).
```



Breast cancer

```
## Compute intercorrelation
data("BreastCancer")
p <- ncol(BreastCancer)-2
sum(cor(sapply(BreastCancer[ , -c(1, 11)], function(x)
  if (!is.numeric(x)) as.numeric(x) else x), use = "pairwise.complete")) / (p*(p-1))
```

```
## [1] 0.7277931
```

```
load(file = "BreastCancer MSE.Rda")
load(file = "BreastCancer acc.Rda")
load(file = "BreastCancer tree_size.Rda")
#boxplot(MSE)
#boxplot(acc)
colMeans(MSE)
```

```
##            tree            BART Born-again tree             N=1             N=5
##      0.04735438      0.03167728      0.04427750              NA      0.04680040
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##      0.04493157              NA      0.04426277      0.04396718              NA
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##      0.04169558      0.04177982              NA      0.04895715      0.04811543
```

```
which.min(colMeans(MSE[ , -2]))
```

```
## N=5, palt=0.25
##             10
```

```
which.min(colMeans(acc[ , -2]))
```

```
## N=10, palt=0.25
##              11
```

```
sapply(MSE, sd)
```

```
##            tree            BART Born-again tree             N=1             N=5
##      0.02246363      0.01097219      0.01523935              NA      0.01561629
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##      0.01400681              NA      0.01591395      0.01425620              NA
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##      0.01121713      0.01195805              NA      0.01220546      0.01095069
```

```
#boxplot(tree_size)
colMeans(tree_size)
```

```
##            tree            BART Born-again tree             N=1             N=5
##           11.06              NA           19.94              NA           11.19
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##           11.42              NA           19.68           18.95              NA
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##           20.58           19.80              NA           20.08           19.01
```

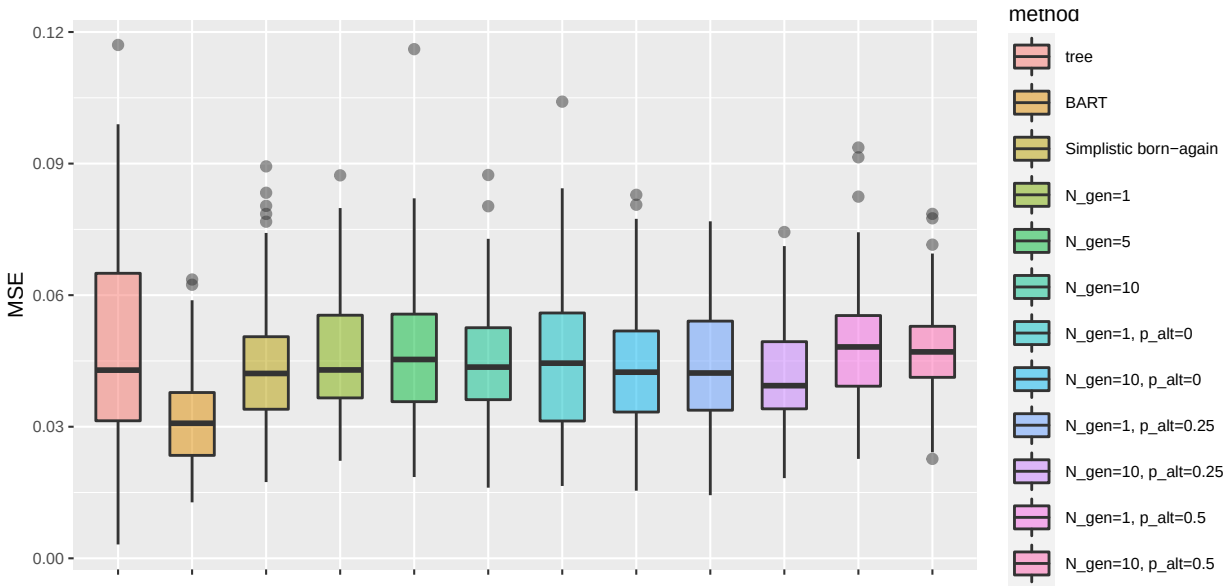
```
sapply(tree_size, sd)
```

```
##            tree            BART Born-again tree             N=1             N=5
##        4.581077              NA        2.411337              NA        1.454321
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##        1.342243              NA        2.428285        2.302283              NA
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##        2.433686        2.605356              NA        2.805766        2.900697
```

```
theme_set(theme_gray(base_size = 8))

## Create long data and plot
omit_inds <- which(names(MSE) %in% c("N=5, palt=0", "N=5, palt=0.25",
                                     "N=5, palt=0.5", "N=5, palt=1"))


MSE_long <- stack(MSE[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(MSE_long) <- c("MSE", "method")
#MSE_long$panel <- ifelse(grepl("BaBayes", MSE_long$method), "Posterior sampling of y",
```

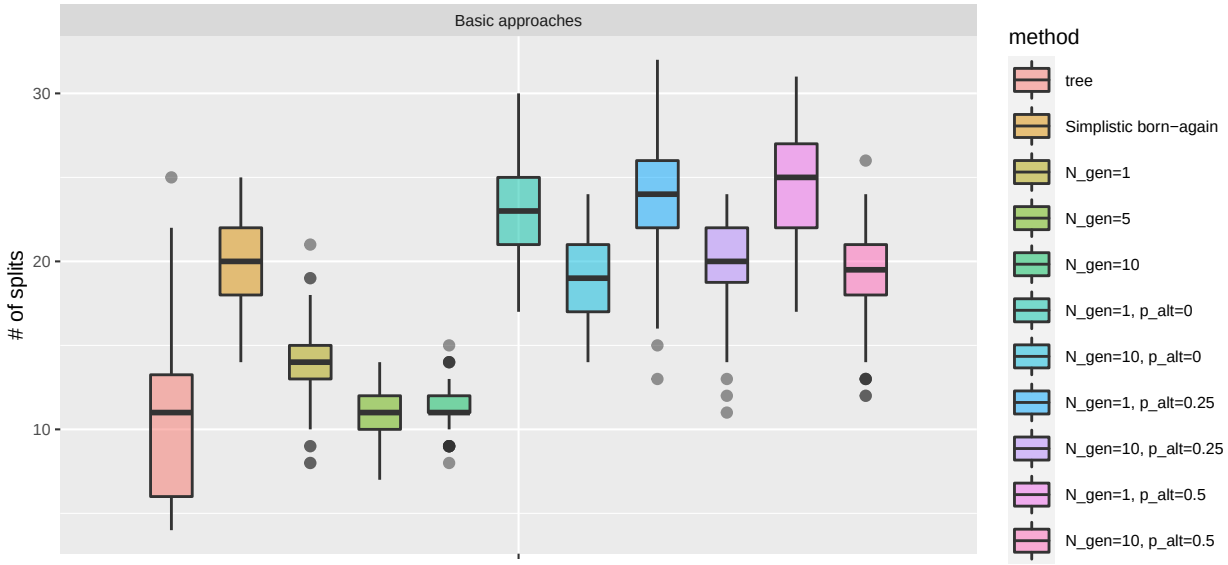
```
#                             ifelse(grepl("BaSmear", MSE_long$method), "Smearing of X", "Basic approach
levs <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10",
                              "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                              "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                              "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                              "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
levels(MSE_long$method) <- levs
ggplot(MSE_long) +
  geom_boxplot(aes(x=method, y=MSE, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  #facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "MSE") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```

```
## Warning: Removed 15 rows containing non-finite values (stat_boxplot).
```




```
size_long <- stack(tree_size[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(size_long) <- c("size", "method")
size_long$panel <- ifelse(grepl("BaBayes", size_long$method), "Posterior sampling",
                          ifelse(grepl("BaSmear", size_long$method) & grepl("BaSmear", size_long$metho
levels(size_long$method) <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10"
                              "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                              "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                              "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                              "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
ggplot(size_long) +
  geom_boxplot(aes(x=panel, y=size, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "# of splits") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```

```
## Warning: Removed 115 rows containing non-finite values (stat_boxplot).
```



Ionosphere

```
library("mlbench")

## Compute intercorrelation
data("Ionosphere")
p <- ncol(Ionosphere)-2
sum(cor(sapply(Ionosphere[ , -c(2L, 35L)], function(x)
  if (!is.numeric(x)) as.numeric(x) else x), use = "pairwise.complete")) / (p*(p-1))
```

```
## [1] 0.1491748
```

```
load(file = "Ionosphere MSE.Rda")
load(file = "Ionosphere acc.Rda")
load(file = "Ionosphere tree_size.Rda")
#boxplot(MSE)
#boxplot(acc)
colMeans(MSE)
```

```
##            tree            BART Born-again tree             N=1             N=5
##      0.08687282      0.07422132      0.09323717      0.10437273      0.10375217
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##      0.09948669      0.09944799      0.09605123      0.09387876      0.11116266
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##      0.10825918      0.11029849      0.12843295      0.12609946      0.12708126
```

```
which.min(colMeans(MSE[ , -2]))
```

```
## tree
##    1
```

```
which.min(colMeans(acc[ , -2]))
```

```
## N=10, palt=0.5
##             14
```

```
sapply(MSE, sd)
```

```
##            tree            BART Born-again tree             N=1             N=5
##      0.03745643      0.01740475      0.02455141      0.02710168      0.02448745
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##      0.02477888      0.02718709      0.02339188      0.02548563      0.02496218
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##      0.02156075      0.02260041      0.02265609      0.02256287      0.02334652
```

```
#boxplot(tree_size)
colMeans(tree_size, na.rm=TRUE)
```

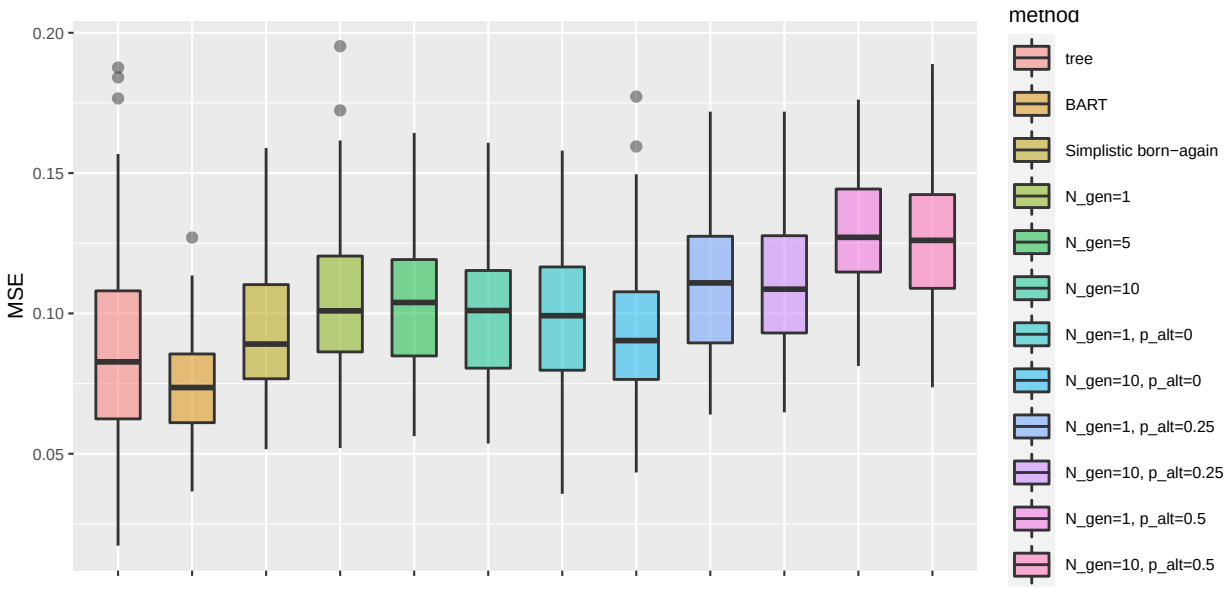
```
##            tree            BART Born-again tree             N=1             N=5
##            7.81             NaN           12.83            6.52            5.17
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##            4.92           14.33           12.71           12.49            9.16
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##            6.36            6.12            8.28            6.38            6.31
```

```
sapply(tree_size, sd)
```

```
##            tree            BART Born-again tree             N=1             N=5
##       2.6041730              NA       1.4978436       1.6906158       0.8415354
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##       0.7872725       2.7672508       1.8764624       1.6907054       2.0583410
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##       1.2187094       1.1658439       1.7528044       1.0519823       1.1164066
```

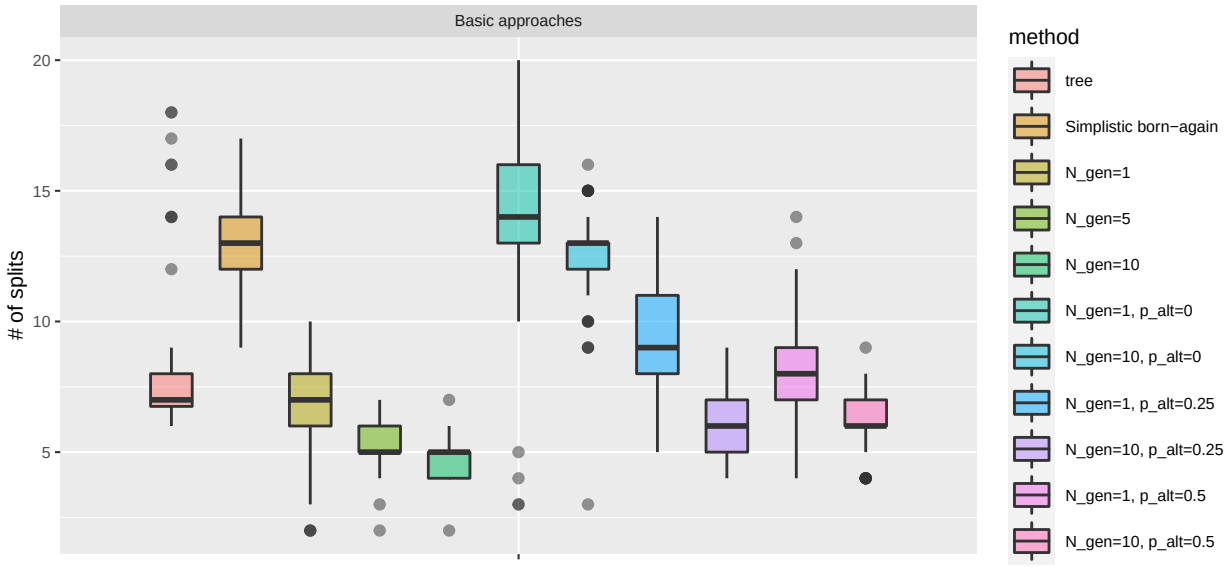
```
theme_set(theme_gray(base_size = 8))

## Create long data and plot
omit_inds <- which(names(MSE) %in% c("N=5, palt=0", "N=5, palt=0.25",
                                     "N=5, palt=0.5", "N=5, palt=1"))


MSE_long <- stack(MSE[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(MSE_long) <- c("MSE", "method")
#MSE_long$panel <- ifelse(grepl("BaBayes", MSE_long$method), "Posterior sampling of y",
#                         ifelse(grepl("BaSmear", MSE_long$method), "Smearing of X", "Basic approach

levs <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10",
```

```
                              "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                              "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                              "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                              "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
levels(MSE_long$method) <- levs
ggplot(MSE_long) +
  geom_boxplot(aes(x=method, y=MSE, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  #facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "MSE") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```




```
size_long <- stack(tree_size[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(size_long) <- c("size", "method")
size_long$panel <- ifelse(grepl("BaBayes", size_long$method), "Posterior sampling",
                          ifelse(grepl("BaSmear", size_long$method) & grepl("BaSmear", size_long$metho
levels(size_long$method) <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10"
                              "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                              "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                              "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                              "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
ggplot(size_long) +
  geom_boxplot(aes(x=panel, y=size, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "# of splits") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```

```
## Warning: Removed 100 rows containing non-finite values (stat_boxplot).
```



Sonar

```
## Compute intercorrelation
data("Sonar")
p <- ncol(Sonar)-1
sum(cor(sapply(Sonar[ , -61L], function(x)
  if (!is.numeric(x)) as.numeric(x) else x))) / (p*(p-1))
```

```
## [1] 0.1166146
```

```
load(file = "Sonar MSE.Rda")
load(file = "Sonar acc.Rda")
load(file = "Sonar tree_size.Rda")
#boxplot(MSE)
#boxplot(acc)
colMeans(MSE)
```

```
##            tree            BART Born-again tree             N=1             N=5
##       0.2141351       0.1468447       0.1847915       0.1998829       0.2053185
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##       0.2060179       0.1911827       0.1832554       0.1832785       0.2019839
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##       0.2142539       0.2139775       0.2144150       0.2215831       0.2230961
```

```
which.min(colMeans(MSE[ , -2]))
```

```
## N=5, palt=0
##           7
```

```r
which.min(colMeans(acc[ , -2]))
```

```
## N=10, palt=0
##            8
```

```r
sapply(MSE, sd)
```

```
##            tree            BART Born-again tree            N=1            N=5
##      0.05597754      0.02335226      0.03209856      0.03057963      0.02687019
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##      0.02553597      0.03340737      0.03102662      0.03316581      0.02696557
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##      0.02292906      0.02112953      0.01913660      0.01589785      0.01430024
```

```r
#boxplot(tree_size)
colMeans(tree_size, na.rm=TRUE)
```

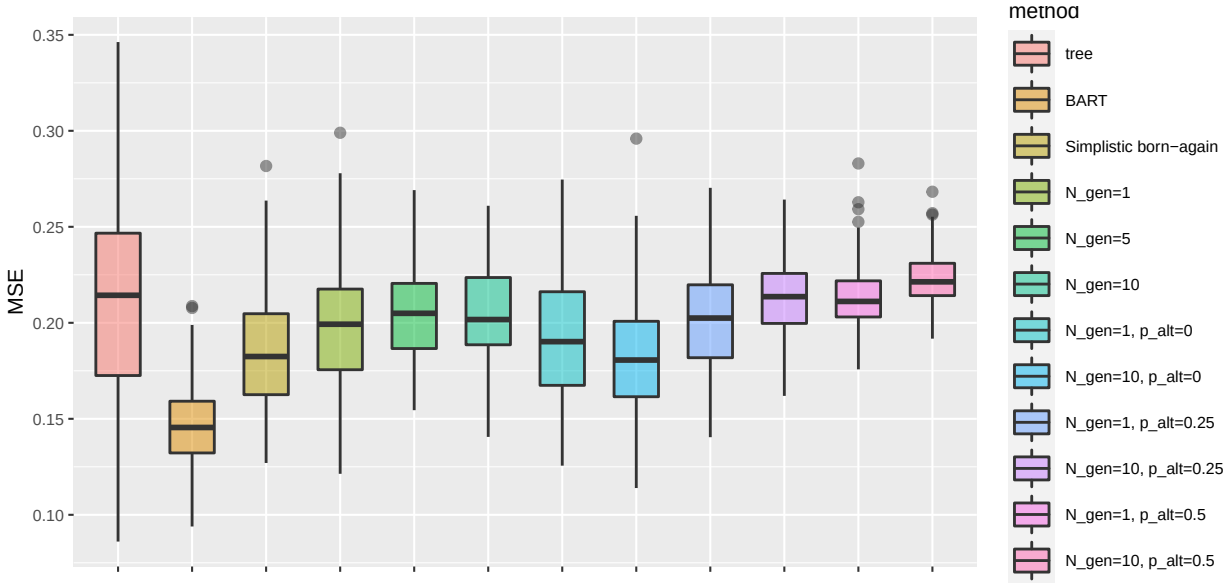
```
##            tree            BART Born-again tree            N=1            N=5
##            4.32             NaN            7.17           4.29           2.48
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##            2.34            8.30            7.06           7.44           5.31
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##            2.82            2.54            3.64           1.91           1.69
```

```r
sapply(tree_size, sd)
```

```
##            tree            BART Born-again tree            N=1            N=5
##       1.7516515              NA       1.3857500       0.9877533       0.8584694
##            N=10     N=1, palt=0     N=5, palt=0    N=10, palt=0  N=1, palt=0.25
##       0.8787043       1.3521401       1.2698525       1.1833867       1.1780398
##  N=5, palt=0.25 N=10, palt=0.25   N=1, palt=0.5   N=5, palt=0.5  N=10, palt=0.5
##       0.6416519       0.6878454       1.0873004       0.6210939       0.6145541
```

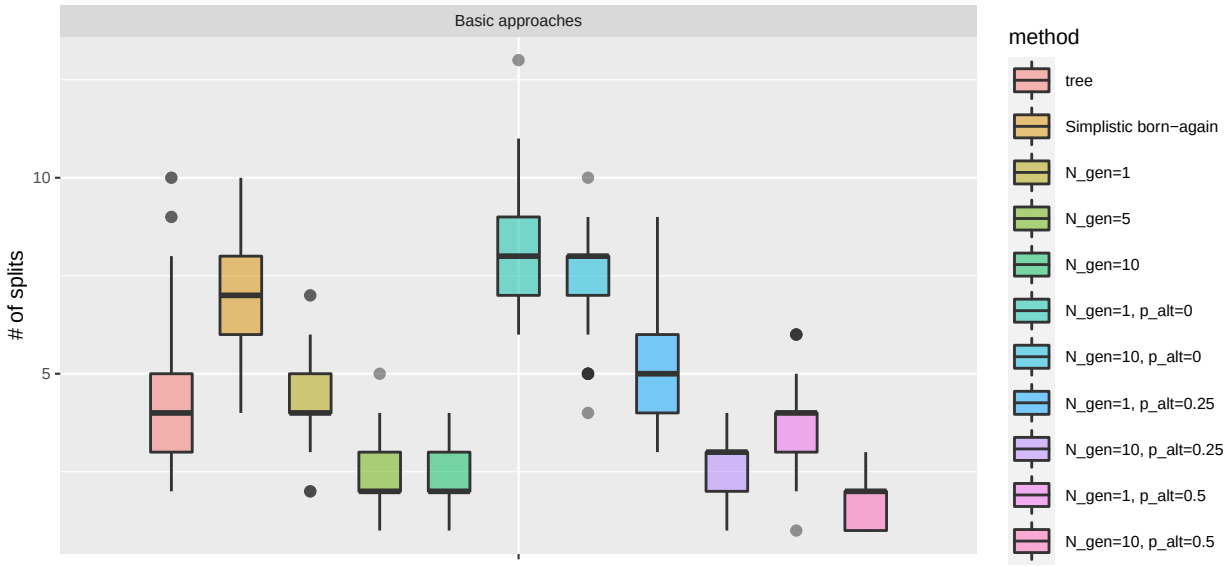
```r
theme_set(theme_gray(base_size = 8))

## Create long data and plot
omit_inds <- which(names(MSE) %in% c("N=5, palt=0", "N=5, palt=0.25",
                                     "N=5, palt=0.5", "N=5, palt=1"))


MSE_long <- stack(MSE[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(MSE_long) <- c("MSE", "method")
#MSE_long$panel <- ifelse(grepl("BaBayes", MSE_long$method), "Posterior sampling of y",
#                         ifelse(grepl("BaSmear", MSE_long$method), "Smearing of X", "Basic approach

levs <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10",
                                "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                                "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                                "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
```

```
                                "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
levels(MSE_long$method) <- levs
ggplot(MSE_long) +
  geom_boxplot(aes(x=method, y=MSE, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  #facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "MSE") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```




```
size_long <- stack(tree_size[ , -omit_inds])
## Create identifier for: original, BA with N, Bayesian BA, Breimand BA
names(size_long) <- c("size", "method")
size_long$panel <- ifelse(grepl("BaBayes", size_long$method), "Posterior sampling",
                          ifelse(grepl("BaSmear", size_long$method) & grepl("BaSmear", size_long$metho
levels(size_long$method) <- c("tree", "BART", "Simplistic born-again", "N_gen=1", "N_gen=5", "N_gen=10"
                              "N_gen=1, p_alt=0", "N_gen=10, p_alt=0",
                              "N_gen=1, p_alt=0.25",  "N_gen=10, p_alt=0.25",
                              "N_gen=1, p_alt=0.5", "N_gen=10, p_alt=0.5",
                              "N_gen=1, p_alt=1", "N_gen=10, p_alt=1")
ggplot(size_long) +
  geom_boxplot(aes(x=panel, y=size, fill = method),
               position=position_dodge(1), alpha = .5, width = .6) +
  facet_grid(~panel, scales = "free", space = "free") +
  labs(x = "", y = "# of splits") +
  theme(axis.title.x=element_blank(), axis.text.x=element_blank())
```

```
## Warning: Removed 100 rows containing non-finite values (stat_boxplot).
```


Basic approaches
# of splits
10
5
method
tree
Simplistic born−again
N_gen=1
N_gen=5
N_gen=10
N_gen=1, p_alt=0
N_gen=10, p_alt=0
N_gen=1, p_alt=0.25
N_gen=10, p_alt=0.25
N_gen=1, p_alt=0.5
N_gen=10, p_alt=0.5