Estimating Variance of Simple Defined Variable Main and Low-Order Interaction Effects

Felix Kapulla

```
knitr::opts_chunk$set(fig.width=14, fig.height=8)
```

```
library(Matrix)
library(tidyverse)
library(ggplot2)
library(ggpubr)
library(ranger)
library(MixMatrix)
library(mvtnorm)
library(stringr)
library(parallel)

cores <- detectCores()
clust <- makeCluster(4)

source('C:/Users/feix_/iCloudDrive/Studium Master/CQM - Thesis Internship/Thesis-VariableEffects/Baseli

parallel::clusterEvalQ(clust,
                       expr = {source('C:/Users/feix_/iCloudDrive/Studium Master/CQM - Thesis Internshi
```

Simulation

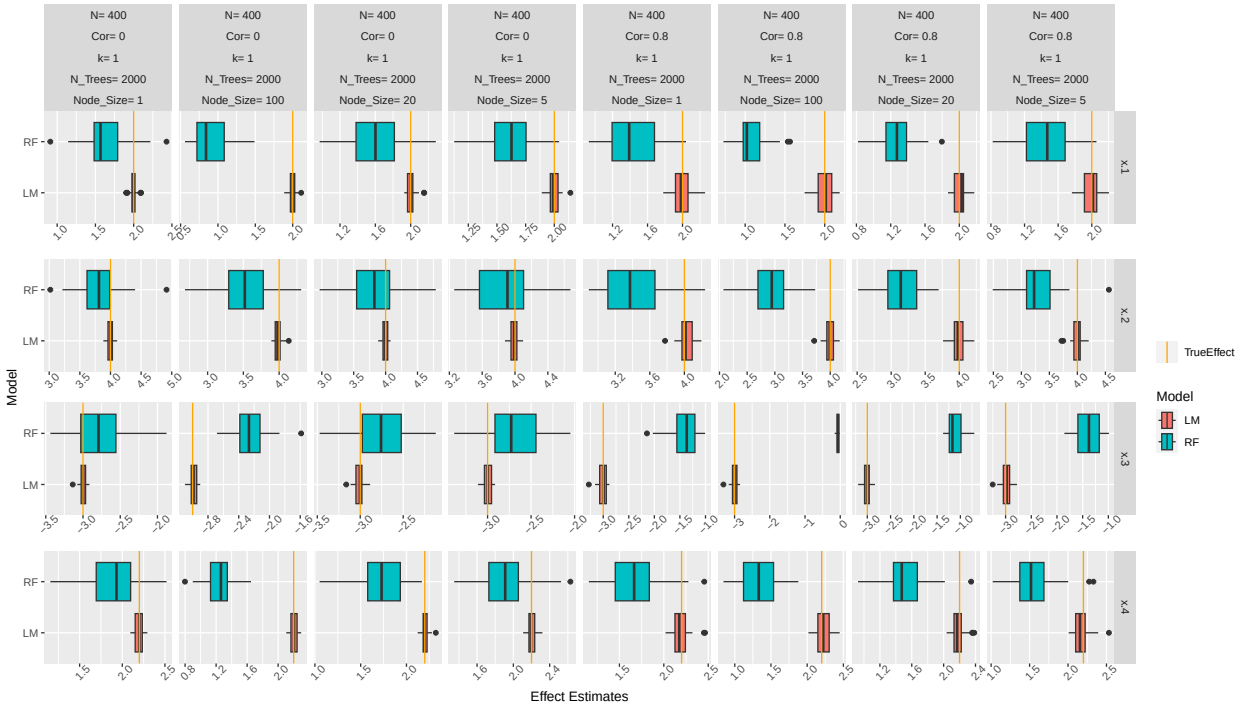
```
n <- c(400) ; num.trees <- 2000 ; repeats <- 30; cor <- c(0, 0.8)
k <- c(1); node_size <- c(1, 5, 20, 100); pdp <- T; ale <- F

formulas <- c("2*x.1+4*x.2-3*x.3+2.2*x.4")
#parallel::clusterExport(cl = clust, varlist = 'formulas')
scenarios <- data.frame(expand.grid(n, num.trees, formulas, repeats,
                                    cor, k, node_size, pdp, ale))
colnames(scenarios) = c("N", "N_Trees", "Formula", "Repeats",
                        "Correlation", "k", "Node_Size", "pdp", "ale")
scenarios$k_idx <- (scenarios$k == unique(scenarios$k)[1])
scenarios[,"Formula"] <- as.character(scenarios[,"Formula"]) ### Formula became Factor
scenarios <- split(scenarios, seq(nrow(scenarios)))
#Run Simulation
system.time(result <- parLapply(cl = clust,
                                X = scenarios,
                                fun = sim_multi))
```

```
##    user  system elapsed
##    0.02    0.39 1767.86
```

```
if (!pdp) {
 print_results(result)
}
effect_plots <- plot_effects(result)
```

```
## `summarise()` has grouped output by 'N', 'cor', 'k', 'num.trees', 'node_size'.
## You can override using the `.groups` argument.
```

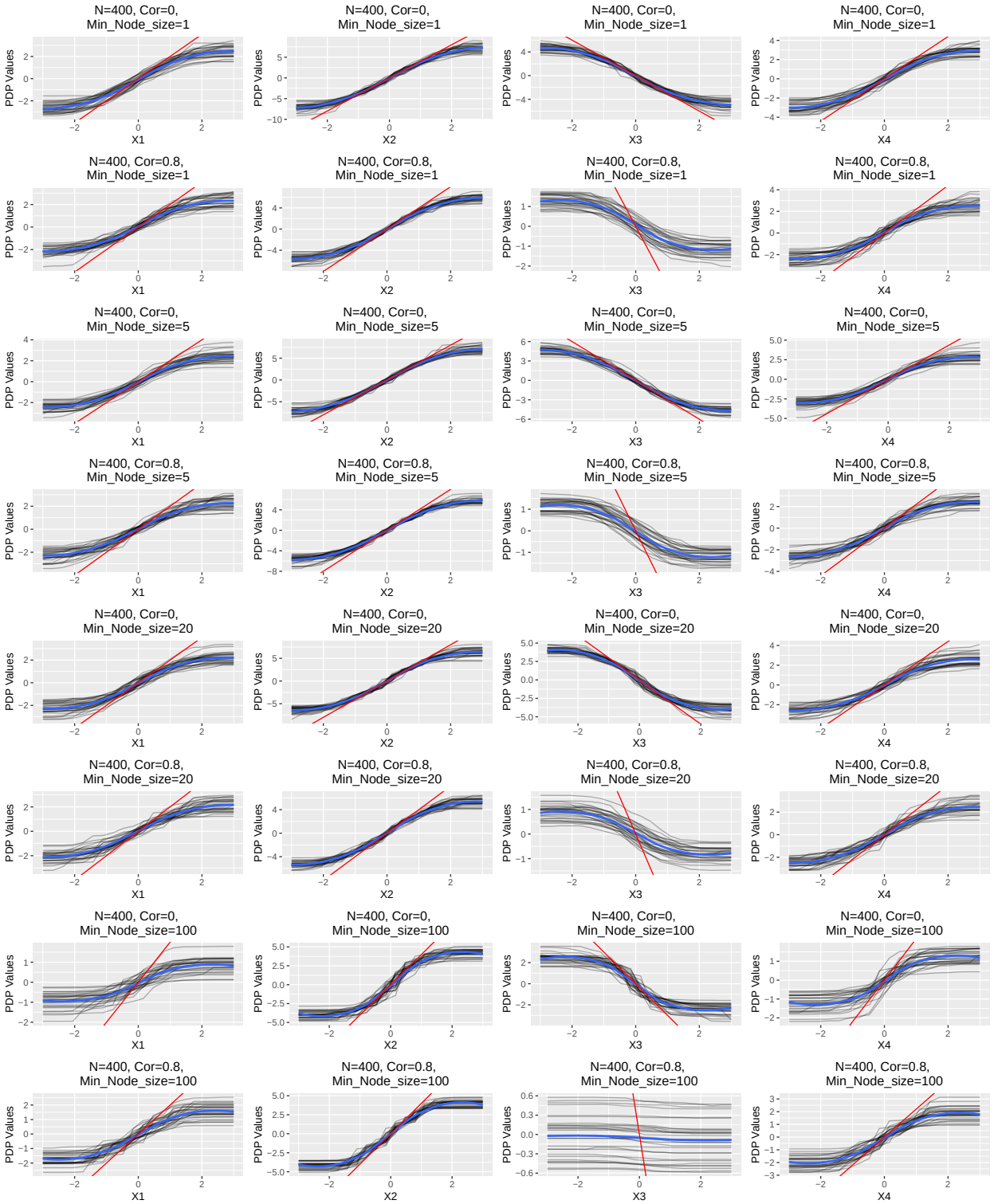
```
#se_plot <- plot_se(result)
effect_plots
```




```
#se_plot
```

```
if (pdp | ale) {
  plot_marginal(result)
}
```

```
## `summarise()` has grouped output by 'N', 'cor', 'k', 'num.trees', 'node_size'.
## You can override using the `.groups` argument.
```




```
n <- c(400) ; num.trees <- 2000 ; repeats <- 30; cor <- c(0, 0.8)
k <- c(1); node_size <- c(1, 5, 20, 100); pdp <- F; ale <- T

formulas <- c("2*x.1+4*x.2-3*x.3+2.2*x.4")
#parallel::clusterExport(cl = clust, varlist = 'formulas')
```

```
scenarios <- data.frame(expand.grid(n, num.trees, formulas, repeats,
                                    cor, k, node_size, pdp, ale))
colnames(scenarios) = c("N", "N_Trees", "Formula", "Repeats",
                        "Correlation", "k", "Node_Size", "pdp", "ale")
scenarios$k_idx <- (scenarios$k == unique(scenarios$k)[1])
scenarios[,"Formula"] <- as.character(scenarios[,"Formula"]) ### Formula became Factor
scenarios <- split(scenarios, seq(nrow(scenarios)))
#Run Simulation
system.time(result <- parLapply(cl = clust,
                                X = scenarios,
                                fun = sim_multi))
```

```
##     user   system  elapsed
##     0.11     0.25 23263.79
```

```
if (!pdp | !ale) {
 print_results(result)
}
```

```
## Setting 1: N = 400 ; k = 1 N_Trees = 2000 ; Correlation = 0 ; Minimum Node Size = 1 ;
##          Formula = 2*x.1+4*x.2-3*x.3+2.2*x.4 ; N_Trees = 2000
## Mean(s) of simulated RF Variable Effect(s):
##   1.590457 3.759162 -2.731272 1.898934
## Mean(s) of simulated LM Variable Effect(s):
##   1.99358 4.006613 -2.992586 2.197358
## True Variable Effect(s):
##   2 4 -3 2.2
## Standard Error of simulated Variable Effects (RF):
##   0.274858 0.3569993 0.4342083 0.2931948 .
## Mean of Standard Errors Estimates of Variable Effects (RF):
##   0.3184091 0.5017334 0.3616917 0.3352094 .
## Number of Smaller Nulls:
##   2 0 1 3
##
## Setting 2: N = 400 ; k = 1 N_Trees = 2000 ; Correlation = 0.8 ; Minimum Node Size = 1 ;
##          Formula = 2*x.1+4*x.2-3*x.3+2.2*x.4 ; N_Trees = 2000
## Mean(s) of simulated RF Variable Effect(s):
##   1.436884 3.473899 -1.453317 1.61732
## Mean(s) of simulated LM Variable Effect(s):
##   1.991534 4.016559 -3.024536 2.190283
## True Variable Effect(s):
##   2 4 -3 2.2
## Standard Error of simulated Variable Effects (RF):
##   0.2943379 0.3806729 0.2269154 0.3772562 .
## Mean of Standard Errors Estimates of Variable Effects (RF):
##   0.3304613 0.3918578 0.2958168 0.3734831 .
## Number of Smaller Nulls:
##   1 0 0 0
##
## Setting 3: N = 400 ; k = 1 N_Trees = 2000 ; Correlation = 0 ; Minimum Node Size = 5 ;
##          Formula = 2*x.1+4*x.2-3*x.3+2.2*x.4 ; N_Trees = 2000
## Mean(s) of simulated RF Variable Effect(s):
##   1.513849 3.807534 -2.640818 1.86278
```

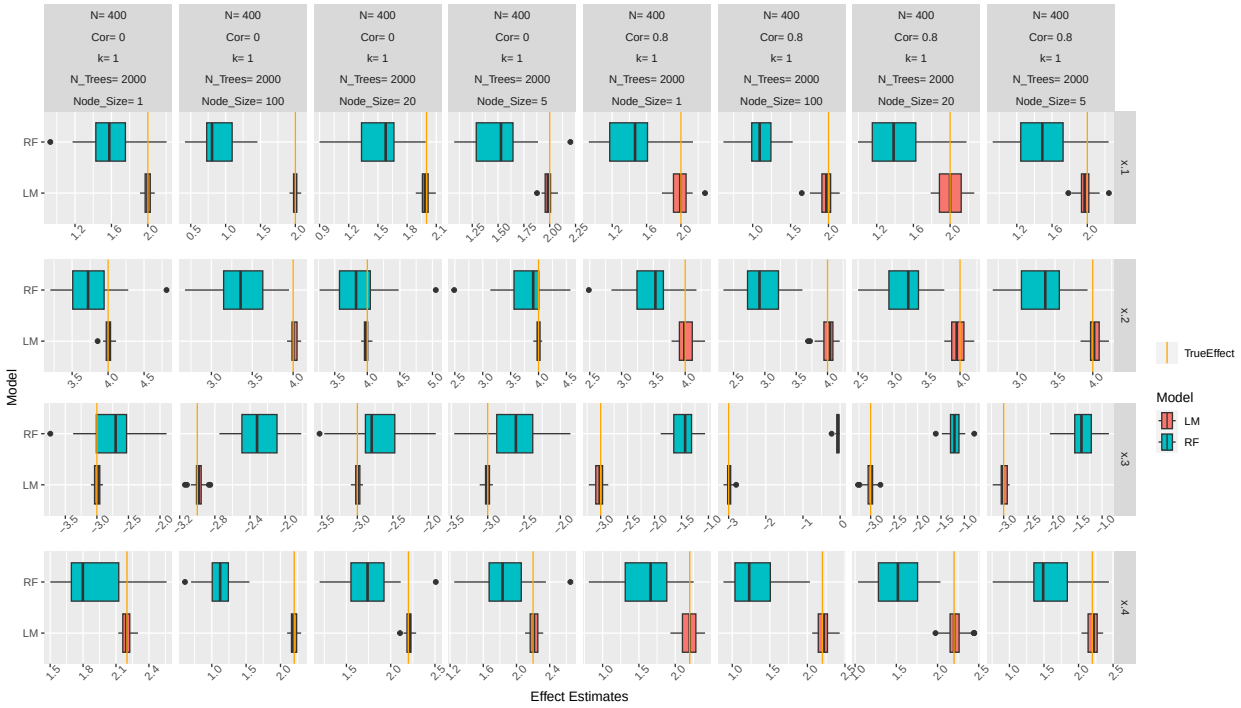
```
## Mean(s) of simulated LM Variable Effect(s):
##   1.986105 3.995492 -3.007327 2.212243
## True Variable Effect(s):
##   2 4 -3 2.2
## Standard Error of simulated Variable Effects (RF):
##   0.2606738 0.4200941 0.4439274 0.32377 .
## Mean of Standard Errors Estimates of Variable Effects (RF):
##   0.3254876 0.4601246 0.3992189 0.3696216 .
## Number of Smaller Nulls:
##   1 0 0 0
##
## Setting 4: N = 400 ; k = 1 N_Trees = 2000 ; Correlation = 0.8 ; Minimum Node Size = 5 ;
##          Formula = 2*x.1+4*x.2-3*x.3+2.2*x.4 ; N_Trees = 2000
## Mean(s) of simulated RF Variable Effect(s):
##   1.488045 3.327135 -1.403397 1.539936
## Mean(s) of simulated LM Variable Effect(s):
##   1.982483 4.01453 -3.009344 2.205627
## True Variable Effect(s):
##   2 4 -3 2.2
## Standard Error of simulated Variable Effects (RF):
##   0.335167 0.3420063 0.2744448 0.4185724 .
## Mean of Standard Errors Estimates of Variable Effects (RF):
##   0.3576032 0.3654929 0.2657965 0.3629202 .
## Number of Smaller Nulls:
##   0 0 1 0
##
## Setting 5: N = 400 ; k = 1 N_Trees = 2000 ; Correlation = 0 ; Minimum Node Size = 20 ;
##          Formula = 2*x.1+4*x.2-3*x.3+2.2*x.4 ; N_Trees = 2000
## Mean(s) of simulated RF Variable Effect(s):
##   1.514051 3.841913 -2.702804 1.737177
## Mean(s) of simulated LM Variable Effect(s):
##   1.986157 3.978554 -2.994653 2.206921
## True Variable Effect(s):
##   2 4 -3 2.2
## Standard Error of simulated Variable Effects (RF):
##   0.2656506 0.3928482 0.3876002 0.2644646 .
## Mean of Standard Errors Estimates of Variable Effects (RF):
##   0.3181252 0.4433591 0.4153515 0.332371 .
## Number of Smaller Nulls:
##   1 0 0 0
##
## Setting 6: N = 400 ; k = 1 N_Trees = 2000 ; Correlation = 0.8 ; Minimum Node Size = 20 ;
##          Formula = 2*x.1+4*x.2-3*x.3+2.2*x.4 ; N_Trees = 2000
## Mean(s) of simulated RF Variable Effect(s):
##   1.428461 3.167285 -1.210037 1.535382
## Mean(s) of simulated LM Variable Effect(s):
##   2.007889 3.976314 -3.00992 2.212362
## True Variable Effect(s):
##   2 4 -3 2.2
## Standard Error of simulated Variable Effects (RF):
##   0.3197697 0.343646 0.1659954 0.2901714 .
## Mean of Standard Errors Estimates of Variable Effects (RF):
##   0.3159151 0.3875408 0.2404706 0.299039 .
## Number of Smaller Nulls:
```

```
##   0 0 0 0
##
## Setting 7: N = 400 ; k = 1 N_Trees = 2000 ; Correlation = 0 ; Minimum Node Size = 100 ;
##          Formula = 2*x.1+4*x.2-3*x.3+2.2*x.4 ; N_Trees = 2000
## Mean(s) of simulated RF Variable Effect(s):
##   0.8745835 3.373534 -2.295488 1.109709
## Mean(s) of simulated LM Variable Effect(s):
##   1.998333 4.009302 -2.983916 2.196428
## True Variable Effect(s):
##   2 4 -3 2.2
## Standard Error of simulated Variable Effects (RF):
##   0.2414336 0.3601595 0.2740612 0.2264973 .
## Mean of Standard Errors Estimates of Variable Effects (RF):
##   0.2763958 0.4586325 0.3623179 0.2745331 .
## Number of Smaller Nulls:
##   0 0 0 1
##
## Setting 8: N = 400 ; k = 1 N_Trees = 2000 ; Correlation = 0.8 ; Minimum Node Size = 100 ;
##          Formula = 2*x.1+4*x.2-3*x.3+2.2*x.4 ; N_Trees = 2000
## Mean(s) of simulated RF Variable Effect(s):
##   1.099816 2.979003 -0.06283446 1.259315
## Mean(s) of simulated LM Variable Effect(s):
##   1.97 4.001176 -2.988286 2.208882
## True Variable Effect(s):
##   2 4 -3 2.2
## Standard Error of simulated Variable Effects (RF):
##   0.2271669 0.325929 0.0456392 0.2913899 .
## Mean of Standard Errors Estimates of Variable Effects (RF):
##   0.2793001 0.407651 0.03413884 0.2671634 .
## Number of Smaller Nulls:
##   0 0 8 2
```

```
effect_plots <- plot_effects(result)
```

```
## `summarise()` has grouped output by 'N', 'cor', 'k', 'num.trees', 'node_size'.
## You can override using the `.groups` argument.
```

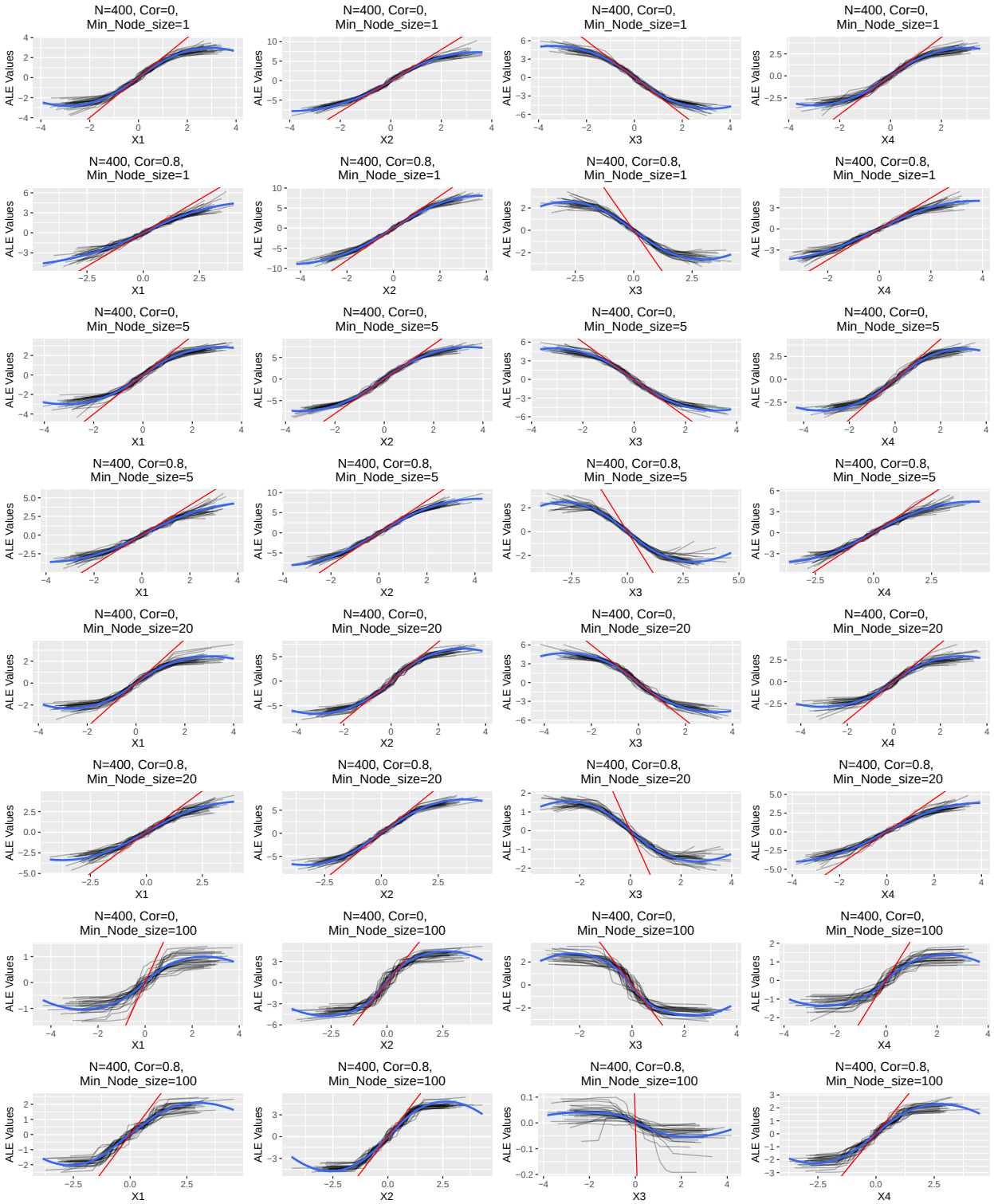
```
#se_plot <- plot_se(result)
effect_plots
```




```
#se_plot
```

```
if (pdp | ale) {
  plot_marginal(result)
}
```

```
## `summarise()` has grouped output by 'N', 'cor', 'k', 'num.trees', 'node_size'.
## You can override using the `.groups` argument.
```


N=400, Cor=0, Min_Node_size=1
N=400, Cor=0, Min_Node_size=1
N=400, Cor=0, Min_Node_size=1
N=400, Cor=0, Min_Node_size=1
N=400, Cor=0.8, Min_Node_size=1
N=400, Cor=0.8, Min_Node_size=1
N=400, Cor=0.8, Min_Node_size=1
N=400, Cor=0.8, Min_Node_size=1
N=400, Cor=0, Min_Node_size=5
N=400, Cor=0, Min_Node_size=5
N=400, Cor=0, Min_Node_size=5
N=400, Cor=0, Min_Node_size=5
N=400, Cor=0.8, Min_Node_size=5
N=400, Cor=0.8, Min_Node_size=5
N=400, Cor=0.8, Min_Node_size=5
N=400, Cor=0.8, Min_Node_size=5
N=400, Cor=0, Min_Node_size=20
N=400, Cor=0, Min_Node_size=20
N=400, Cor=0, Min_Node_size=20
N=400, Cor=0, Min_Node_size=20
N=400, Cor=0.8, Min_Node_size=20
N=400, Cor=0.8, Min_Node_size=20
N=400, Cor=0.8, Min_Node_size=20
N=400, Cor=0.8, Min_Node_size=20
N=400, Cor=0, Min_Node_size=100
N=400, Cor=0, Min_Node_size=100
N=400, Cor=0, Min_Node_size=100
N=400, Cor=0, Min_Node_size=100
N=400, Cor=0.8, Min_Node_size=100
N=400, Cor=0.8, Min_Node_size=100
N=400, Cor=0.8, Min_Node_size=100
N=400, Cor=0.8, Min_Node_size=100
ALE Values
X1
X2
X3
X4