Random Forest Regression

Results based on data-specific hyper-parameter tuning

Amos Okutse

22 December, 2022

Contents

0.0.1 PART A: FULL DATA 1
0.0.2 PART B: OBSERVED DATA ONLY 2
0.0.3 PART C: MODIFIED AS IN PART B WITH PREDICTIONS FOR EVERYONE . . 3
0.1 TABLE OF RANDOM FOREST RESULTS 6

```
rm(list = ls())
## load the saved single data files
load("C:\\Users\\aokutse\\OneDrive - Brown
 ↪  University\\ThesisResults\\data\\df_one.RData")
load("C:\\Users\\aokutse\\OneDrive - Brown
 ↪  University\\ThesisResults\\data\\df_two.RData")
load("C:\\Users\\aokutse\\OneDrive - Brown
 ↪  University\\ThesisResults\\data\\df_three.RData")
load("C:\\Users\\aokutse\\OneDrive - Brown
 ↪  University\\ThesisResults\\data\\df_four.RData")


## load the saved list data files
load("C:\\Users\\aokutse\\OneDrive - Brown
 ↪  University\\ThesisResults\\data\\dsets1.RData")
load("C:\\Users\\aokutse\\OneDrive - Brown
 ↪  University\\ThesisResults\\data\\dsets2.RData")
load("C:\\Users\\aokutse\\OneDrive - Brown
 ↪  University\\ThesisResults\\data\\dsets3.RData")
load("C:\\Users\\aokutse\\OneDrive - Brown
 ↪  University\\ThesisResults\\data\\dsets4.RData")
```

0.0.1 PART A: FULL DATA

```
## create the function to return the desired estimates from the model
rf_model_one <- function(df = NULL, mtry = NULL, min_n = NULL){
  # fit random forest model for all individuals
  rf_all <- rand_forest(trees = 1000, mtry = mtry, min_n = min_n) %>%
```

```
  set_mode("regression") %>%
  set_engine("ranger") %>%
  fit(formula = y ~ A + x1 + x2 + x3 + x4, data = df)
## set A = 0 and generate predictions for everyone
  df_A0 <- df
  df_A0$A <- 0
  pred_A0 <- predict(rf_all, df_A0)
## set A = 1 and generate predictions for everyone
  df_A1 <- df
  df_A1$A <- 1
  pred_A1 <- predict(rf_all, df_A1)
## compute the ATE
  ATE_adjusted = mean(pred_A1$.pred - pred_A0$.pred)
## compute the bias
  bias_adjusted = ATE_adjusted - 50
## return the results as a data frame
  rslt = data.frame("ATE_adjusted" = ATE_adjusted, "bias_adjusted" = bias_adjusted)
  return(rslt)
}
```

```
# combine the results into a data frame
  rf_onea <- onea %>% map_dfr(data.frame)
  rf_oneb <- oneb %>% map_dfr(data.frame)
  rf_onec <- onec %>% map_dfr(data.frame)
  rf_oned <- oned %>% map_dfr(data.frame)
```

0.0.2 PART B: OBSERVED DATA ONLY

- Analysis restricted on the observed data alone, that is, where $R = 1$. Predictions are then made to only those individuals with observed outcomes.

```
## create the function to return the desired estimates from the model
rf_model_two <- function(df = NULL, mtry = NULL, min_n = NULL){
## filter the data to have only individuals with R = 1
  df = dplyr::filter(df, R == 1)
  # fit random forest model for all individuals with R=1
  rf_two <- rand_forest(trees = 1000, mtry = mtry, min_n = min_n) %>%
  set_mode("regression") %>%
  set_engine("ranger") %>%
  fit(formula = y ~ A + x1 + x2 + x3 + x4, data = df)
## set A=0 and generate predictions for those with R=1
  df_A0 <- df
  df_A0$A <- 0
  pred_A0 <- predict(rf_two, df_A0)
## set A=1 and generate predictions for those with R=1
  df_A1 <- df
  df_A1$A <- 1
  pred_A1 <- predict(rf_two, df_A1)
## compute the ATE
  ATE_adjusted = mean(pred_A1$.pred)-mean(pred_A0$.pred)
## compute the bias
```

```
  bias_adjusted = ATE_adjusted - 50
## return the results as a data frame
  rslt = data.frame("ATE_adjusted" = ATE_adjusted, "bias_adjusted" = bias_adjusted)
  return(rslt)
}
```

```
# combine the results into a data frame
  rf_twoa <- twoa %>% map_dfr(data.frame)
  rf_twob <- twob %>% map_dfr(data.frame)
  rf_twoc <- twoc %>% map_dfr(data.frame)
  rf_twod <- twod %>% map_dfr(data.frame)
```

0.0.3 PART C: MODIFIED AS IN PART B WITH PREDICTIONS FOR EVERYONE

```
## create the function to return the desired estimates from the model
rf_model_three <- function(df = NULL, mtry = NULL, min_n = NULL){
  # fit random forest model for all individuals with R=1
  rf_three <- rand_forest(trees = 1000, mtry = mtry, min_n = min_n) %>%
  set_mode("regression") %>%
  set_engine("ranger") %>%
  fit(formula = y ~ A + x1 + x2 + x3 + x4, data = dplyr::filter(df, R == 1))
## set A = 0 and generate predictions for everyone
  df_A0 <- df
  df_A0$A <- 0
  pred_A0 <- predict(rf_three, df_A0)
## set A = 1 and generate predictions for everyone
  df_A1 <- df
  df_A1$A <- 1
  pred_A1 <- predict(rf_three, df_A1)
## compute the ATE
  ATE_adjusted = mean(pred_A1$.pred) - mean(pred_A0$.pred)
## compute the bias
  bias_adjusted = ATE_adjusted - 50
## return the results as a data frame
  rslt = data.frame("ATE_adjusted" = ATE_adjusted, "bias_adjusted" = bias_adjusted)
  return(rslt)
}
```

```
# combine the results into a data frame
  rf_threea <- threea %>% map_dfr(data.frame)
  rf_threeb <- threeb %>% map_dfr(data.frame)
  rf_threec <- threec %>% map_dfr(data.frame)
  rf_threed <- threed %>% map_dfr(data.frame)
```

```
##-------------------------------------------------------------------------------
## case 1 [n = 500, sd = 1]
##-------------------------------------------------------------------------------
## full
```

```
full <- c(n = nrow(df_one), ate = mean(rf_onea$ATE_adjusted), sd =
 ↪  sd(rf_onea$ATE_adjusted), bias = mean(rf_onea$bias_adjusted), sd_bias =
 ↪  sd(rf_onea$bias_adjusted))
full
```

```
##            n          ate           sd         bias      sd_bias
## 500.00000000  49.90483223   1.02460581  -0.09516777   1.02460581
```

```
## observed
obs <- c(n = nrow(subset(df_one, R == 1)), ate = mean(rf_twoa$ATE_adjusted), sd =
 ↪  sd(rf_twoa$ATE_adjusted), bias = mean(rf_twoa$bias_adjusted), sd_bias =
 ↪  sd(rf_twoa$bias_adjusted))
obs
```

```
##           n         ate          sd        bias     sd_bias
## 244.0000000  49.5513099   1.8959442  -0.4486901   1.8959442
```

```
## observed modified
obs_m <- c(n = nrow(subset(df_one, R == 1)), ate = mean(rf_threea$ATE_adjusted), sd =
 ↪  sd(rf_threea$ATE_adjusted), bias = mean(rf_threea$bias_adjusted), sd_bias =
 ↪  sd(rf_threea$bias_adjusted))
obs_m
```

```
##           n         ate          sd        bias     sd_bias
## 244.0000000  49.4899904   2.5483248  -0.5100096   2.5483248
```

```
##-------------------------------------------------------------------------------
## case 2 [n = 500, sd = 45]
##-------------------------------------------------------------------------------
full2 <- c(n = nrow(df_two), ate = mean(rf_oneb$ATE_adjusted), sd =
 ↪  sd(rf_oneb$ATE_adjusted), bias = mean(rf_oneb$bias_adjusted), sd_bias =
 ↪  sd(rf_oneb$bias_adjusted))
full2
```

```
##          n        ate         sd       bias    sd_bias
## 500.000000  48.006756   5.257440  -1.993244   5.257440
```

```
## observed
obs2 <- c(n = nrow(subset(df_two, R == 1)), ate = mean(rf_twob$ATE_adjusted), sd =
 ↪  sd(rf_twob$ATE_adjusted), bias = mean(rf_twob$bias_adjusted), sd_bias =
 ↪  sd(rf_twob$bias_adjusted))
obs2
```

```
##          n        ate         sd       bias    sd_bias
## 258.000000  45.360595   8.788519  -4.639405   8.788519
```

```
## observed modified
obs_m2 <- c(n = nrow(subset(df_two, R == 1)), ate = mean(rf_threeb$ATE_adjusted), sd =
 ↪  sd(rf_threeb$ATE_adjusted), bias = mean(rf_threeb$bias_adjusted), sd_bias =
 ↪  sd(rf_threeb$bias_adjusted))
obs_m2
```

```
##          n        ate         sd       bias    sd_bias
## 258.000000  45.156242   9.569304  -4.843758   9.569304
```

```
##-------------------------------------------------------------------------------
## case 3 [n = 2000, sd = 1]
##-------------------------------------------------------------------------------
full3 <- c(n = nrow(df_three), ate = mean(rf_onec$ATE_adjusted), sd =
 ↪  sd(rf_onec$ATE_adjusted), bias = mean(rf_onec$bias_adjusted), sd_bias =
 ↪  sd(rf_onec$bias_adjusted))
full3
```

```
##             n           ate            sd          bias       sd_bias
## 2000.00000000   49.98983743    0.32279955   -0.01016257    0.32279955
```

```
## observed
obs3 <- c(n = nrow(subset(df_three, R == 1)), ate = mean(rf_twoc$ATE_adjusted), sd =
 ↪  sd(rf_twoc$ATE_adjusted), bias = mean(rf_twoc$bias_adjusted), sd_bias =
 ↪  sd(rf_twoc$bias_adjusted))
obs3
```

```
##           n          ate           sd         bias      sd_bias
## 997.00000000  49.98958552   0.54109767  -0.01041448   0.54109767
```

```
## observed modified
obs_m3 <- c(n = nrow(subset(df_three, R == 1)), ate = mean(rf_threec$ATE_adjusted), sd =
 ↪  sd(rf_threec$ATE_adjusted), bias = mean(rf_threec$bias_adjusted), sd_bias =
 ↪  sd(rf_threec$bias_adjusted))
obs_m3
```

```
##           n          ate           sd         bias      sd_bias
## 997.00000000  49.97462574   0.80434083  -0.02537426   0.80434083
```

```
##-------------------------------------------------------------------------------
## case 4 [n = 2000, sd = 45]
##-------------------------------------------------------------------------------
full4 <- c(n = nrow(df_four), ate = mean(rf_oned$ATE_adjusted), sd =
 ↪  sd(rf_oned$ATE_adjusted), bias = mean(rf_oned$bias_adjusted), sd_bias =
 ↪  sd(rf_oned$bias_adjusted))
full4
```

```
##           n          ate           sd         bias      sd_bias
## 2000.0000000   48.1324842    0.5115253   -1.8675158    0.5115253
```

```
## observed
obs4 <- c(n = nrow(subset(df_four, R == 1)), ate = mean(rf_twod$ATE_adjusted), sd =
 ↪  sd(rf_twod$ATE_adjusted), bias = mean(rf_twod$bias_adjusted), sd_bias =
 ↪  sd(rf_twod$bias_adjusted))
obs4
```

Table 1: Random forest results averaged across n = 1000 datasets under full, observed, and observed modified analysis

Data generating values	n	ate	sd	bias	sd_bias
n = 500, SD = 1	500	49.90483	1.0246058	-0.0951678	1.0246058
n = 500, SD = 1	244	49.55131	1.8959442	-0.4486901	1.8959442
n = 500, SD = 1	244	49.48999	2.5483248	-0.5100096	2.5483248
n = 500, SD = 45	500	48.00676	5.2574402	-1.9932437	5.2574402
n = 500, SD = 45	258	45.36059	8.7885188	-4.6394051	8.7885188
n = 500, SD = 45	258	45.15624	9.5693041	-4.8437580	9.5693041
n = 2000, SD = 1	2000	49.98984	0.3227996	-0.0101626	0.3227996
n = 2000, SD = 1	997	49.98959	0.5410977	-0.0104145	0.5410977
n = 2000, SD = 1	997	49.97463	0.8043408	-0.0253743	0.8043408
n = 2000, SD = 45	2000	48.13248	0.5115253	-1.8675158	0.5115253
n = 2000, SD = 45	1003	49.22469	0.7476698	-0.7753129	0.7476698
n = 2000, SD = 45	1003	49.14068	1.0132343	-0.8593234	1.0132343

```
##            n          ate           sd         bias      sd_bias
## 1003.0000000   49.2246871    0.7476698   -0.7753129    0.7476698
```

```
## observed modified
obs_m4 <- c(n = nrow(subset(df_four, R == 1)), ate = mean(rf_threed$ATE_adjusted), sd =
 ↪  sd(rf_threed$ATE_adjusted), bias = mean(rf_threed$bias_adjusted), sd_bias =
 ↪  sd(rf_threed$bias_adjusted))
obs_m4
```

```
##            n          ate           sd         bias      sd_bias
## 1003.0000000   49.1406766    1.0132343   -0.8593234    1.0132343
```

0.1 TABLE OF RANDOM FOREST RESULTS

```
random_forest2 <- bind_rows(list("n = 500, SD = 1" = full, "n = 500, SD = 1" = obs, "n =
 ↪  500, SD = 1"= obs_m, "n = 500, SD = 45" = full2, "n = 500, SD = 45" = obs2, "n = 500,
 ↪  SD = 45" = obs_m2, "n = 2000, SD = 1" = full3, "n = 2000, SD = 1" = obs3, "n = 2000,
 ↪  SD = 1" = obs_m3, "n = 2000, SD = 45" = full4, "n = 2000, SD = 45"=  obs4, "n = 2000,
 ↪  SD = 45" = obs_m4), .id = "Data generating values")
kable(random_forest2, format = "latex", caption = "Random forest results averaged across
 ↪  n = 1000 datasets under full, observed, and observed modified analysis")
```

```
## the order of the rows starts with n = 500
write.csv(random_forest2, file = "C:\\Users\\aokutse\\OneDrive - Brown
 ↪  University\\ThesisResults\\[4]_random_forest\\final_rf\\rforest_results_two.csv")
```