Missingness simulation

2023-03-29

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(grf)
library(estimatr)
set.seed(60637)

files <- list.files('../data',
                    pattern = '^cleaned-data.*rds$',
                    full.names = TRUE)

(INPUT_FILENAME <- files[which.max(file.info(files)$mtime)])
```

```
## [1] "../data/cleaned-data_2023-03-28.rds"
```

```
df_treat <- readRDS(INPUT_FILENAME)
dfx <- df_treat[which(df_treat$batch == 5),]

dfx$ws_eval <- as.factor(case_when(as.numeric(dfx$W) == 1 ~ 1, # control
                                   as.numeric(dfx$W) == 2 ~  2, # headline factcheck,
                                   as.numeric(dfx$W) == 5 ~ 3, # headline related,
                                   as.numeric(dfx$W) == 6 ~ 4, # respondent accuracy
                                   as.numeric(dfx$W) == 11 ~ 5, # facebook tips,
                                   # combine optimal with small groups
                                   TRUE ~ 6 # other optimal respondent (8/12)
))

# attrition probabilities across
round(prop.table(table(dfx$attrited, dfx$ws_eval), margin =2),3)
```

```
##
##         1     2     3     4     5     6
##   0 0.930 0.926 0.912 0.911 0.892 0.917
##   1 0.070 0.074 0.088 0.089 0.108 0.083
```

```
temp_mat <- aggregate(pre_false ~ attrited + ws_eval,
          data = dfx,
          function(x) c(`estimate` = mean(x), std.error = sd(x)/sqrt(length(x))))
```

```r
# T-test for differences
# false
t.test(dfx$pre_false ~dfx$attrited)
```

```
##
##  Welch Two Sample t-test
##
## data:  dfx$pre_false by dfx$attrited
## t = 1.8575, df = 1201.8, p-value = 0.06349
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -0.005676392  0.207543301
## sample estimates:
## mean in group 0 mean in group 1
##        1.961352        1.860419
```

```r
diff(t.test(dfx$pre_false ~dfx$attrited)$estimate)
```

```
## mean in group 1
##      -0.1009335
```

```r
# true
t.test(dfx$pre_true ~dfx$attrited)
```

```
##
##  Welch Two Sample t-test
##
## data:  dfx$pre_true by dfx$attrited
## t = 3.0352, df = 1188.3, p-value = 0.002456
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  0.05450421 0.25377975
## sample estimates:
## mean in group 0 mean in group 1
##        2.692527        2.538385
```

```r
diff(t.test(dfx$pre_true ~dfx$attrited)$estimate)
```

```
## mean in group 1
##       -0.154142
```

```r
# true data generating process
mmx0 <- model.matrix(Y~pre_false + pre_true + ws_eval, data = dfx[which(dfx$attrited == 0),])[,-1]
forest.Y <- regression_forest(X = mmx0, Y = dfx$Y[which(dfx$attrited == 0)])


# simuluated large data set, from real data
superX <- dfx[sample(1:nrow(dfx), 1e5, replace = TRUE),]
mmxfull <- model.matrix(Y~pre_false + pre_true + ws_eval, data = superX)[,-1]

ys <-   sapply(1:6, function(x){
  if(x!=1){
    mmxfull[,3:7] <- 0
    mmxfull[,1+x] <- 1
  }
  predict(forest.Y, newdata = mmxfull)[,1]
})

ate_truth <- colMeans(ys[,2:6] - ys[,1])

# simulated estimating procedures
niter <- 1e1
out_mat <- list(
  imputed = matrix(NA, nrow = niter, ncol = 5),
  weighted = matrix(NA, nrow = niter, ncol = 5))

for(i in 1:niter){
  newdat <- cbind(ys, superX)[sample(1:nrow(dfx), replace = TRUE),]
  newys <- newdat[,1:6]
  newdat <- newdat |>
    mutate(
      Y = newys[cbind(1:nrow(dfx), newdat$ws_eval)] + rnorm(nrow(dfx)),
      Y_imputed = case_when(
        attrited == 0 ~ Y,
        TRUE ~ - pre_false + 0.5 * pre_true
      )
    )

  uncens_prob <- probability_forest(X = as.matrix(newdat[, c('pre_false', 'pre_true')]),
                                    Y = as.factor(1*(newdat$attrited == 0)))$predictions[which(newdat$a
  balwts_uncens <- (1/uncens_prob)[, 2]

  # imputed
  li <- coef(lm_lin(Y_imputed~ws_eval, data = newdat,
               covariates = formula(' ~ pre_false + pre_true')))[2:6]

  # weighted
  lw <- coef(lm_lin(Y~ws_eval, data = newdat[which(newdat$attrited == 0),],
               weights = balwts_uncens,
               covariates = formula(' ~ pre_false + pre_true')))[2:6]

  out_mat[['imputed']][i,] <- ate_truth-li
  out_mat[['weighted']][i,] <- ate_truth-lw

}

do.call(rbind, lapply(out_mat, function(x) colMeans(x)))
```

```
##                  [,1]         [,2]        [,3]        [,4]        [,5]
## imputed  -0.017446336 -0.001124422 0.007367627  0.01167168  0.13978601
## weighted -0.005343066  0.007142284 0.003823731 -0.00185681 -0.01032685
```

```
lapply(out_mat, function(x) mean(abs(colMeans(x))))
```

```
## $imputed
## [1] 0.03547922
##
## $weighted
## [1] 0.005698548
```