baseline

2022-06-02

DGP

```
n = 100
t1 = 40
t0 = 21
p = 1
mu_a = 0
sig_a = 1
mu_g = 1
sig_g = 1

mc <- 5
tau_xbcf_mc <- array(0, dim = c(mc, n, t1-t0+1))
tau_freq_mc <- array(0, dim = c(mc, t1))
# pct_bias_mc <- array(0, dim = c(mc, n, t1-t0+1))

# alpha = rnorm(n, mu_a, sig_a)
# gamma = rnorm(t1, mu_g, sig_g)

x = as.matrix(rnorm(n))
mu <- function(x, t) {1 + 0.5*x - 0.5*t  + 0.1*x*t}
tau <- function(x){a = (x + 1.5)^2; 5*sqrt(a) + sin(5*a) + 1}
mu_mat <- outer(as.vector(x), 1:t1, mu)
tau_mat <- matrix(0, n, t1-t0+1)
tau_mat[,1] <- tau(x)
for (i in 2:(t1-t0+1)){
  tau_mat[,i] <- 0.9*tau_mat[,i-1]
}

s <- sd(x)
pi <- 0.8*pnorm(3*x / s -0.5*x ) + 0.05 + runif(n)/10
z = rbinom(n, 1, pi)
```

MCMC

```
pb <- txtProgressBar(min = 0, max = mc, initial = 0)
count_iter <- 0
filename <- "longBet_sim_prog.RData"
if (!exists(filename)){
  save(count_iter, tau_mat, file = filename)
}

for (iter in 1:mc){
  setTxtProgressBar(pb,iter)
```

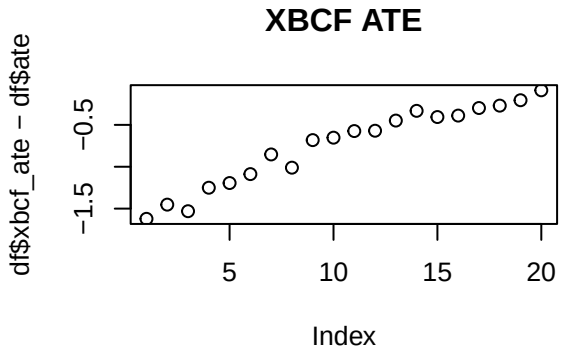
```
  load(filename)
  if (count_iter > iter) {next}

  eps = matrix(rnorm(n*t1, 0, 0.2), nrow = n, ncol = t1)

  y0 = y1 = y = matrix(0, nrow = n, ncol = t1)
  y0 <- mu_mat
  # for (i in 1:n){
  #   y0[i,] = y0[i,] + alpha[i]
  # }
  # for (j in 1:t1){
  #   y0[,j] = y0[,j] + gamma[j]
  # }
  y0 = y0 + eps
  y1 = y0
  y1[, t0:t1] = y0[, t0:t1] + tau_mat
  z_mat = matrix(rep(z, t1), n, t1)
  y = y0 * (1-z_mat) + y1 * z_mat

  # XBCF
  fit <- longBet_prog(y, x, z, t0, 100, 20, 40)
  tau_hat <- colMeans(fit$tauhat)
  # pct_bias <- abs((tau_hat - tau_mat) / tau_mat)
  tau_xbcf_mc[iter,,] <- tau_hat[,t0:t1]

  # Frequentist
  ytilde <- y - matrix(rowMeans(y)) %*% rep(1, t1) - rep(1, n) %*% t(matrix(colMeans(y))) + mean(y)
  recenter <- mean((colMeans(ytilde[matrix(z_mat[,1])==1,]) - colMeans(ytilde[matrix(z_mat[,1])==0,]))[
  tau_freq_mc[iter,] <- colMeans(ytilde[matrix(z_mat[,1])==1,]) - colMeans(ytilde[matrix(z_mat[,1])==0,]

  count_iter <- iter
  save(count_iter, tau_mat, tau_xbcf_mc, tau_freq_mc, file = filename)

}
```

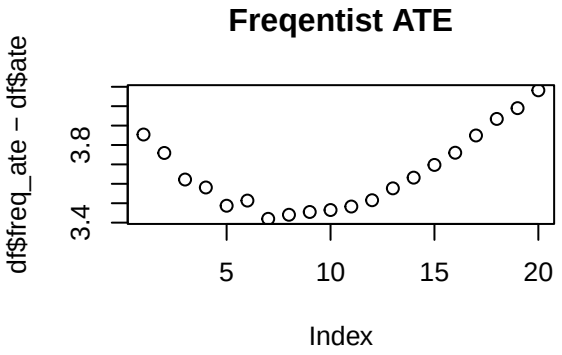
```
## ============================================================================
```

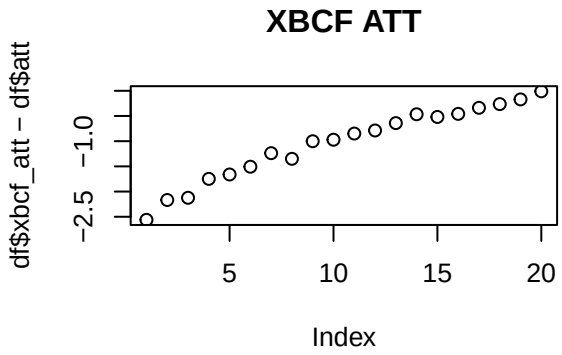
Results

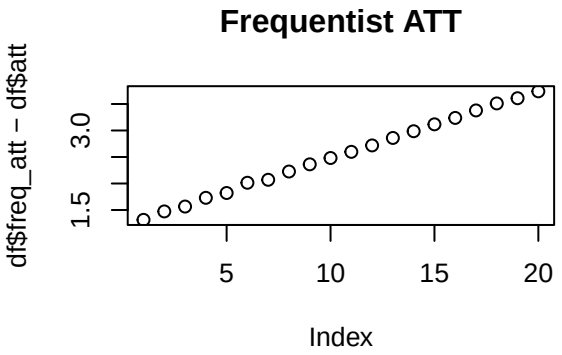
```
load(filename)
df <- data.frame(time = seq(t0,t1),
                 xbcf_ate = colMeans(colMeans(tau_xbcf_mc)),
                 freq_ate = colMeans(tau_freq_mc[,t0:t1]),
                 ate = colMeans(tau_mat),
                 xbcf_att = colMeans(colMeans(tau_xbcf_mc)[z_mat[,1]==1,]),
                 freq_att = colMeans(tau_freq_mc[,t0:t1]),
                 att = colMeans(tau_mat[z_mat[,1]==1,])
                 )
```

```
par(mfrow=c(2,2))
plot(df$xbcf_ate - df$ate, main = "XBCF ATE")
plot(df$freq_ate - df$ate, main = "Freqentist ATE")
plot(df$xbcf_att - df$att, main = "XBCF ATT")
plot(df$freq_att - df$att, main = "Frequentist ATT")
```

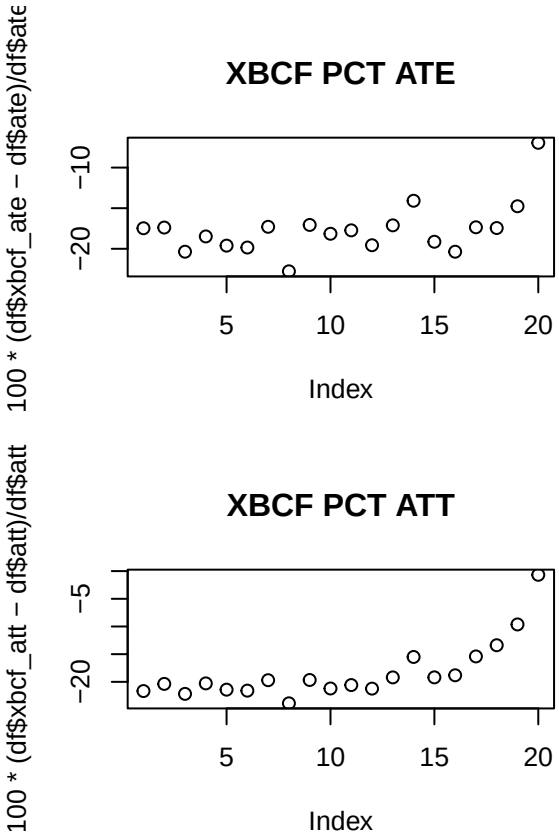


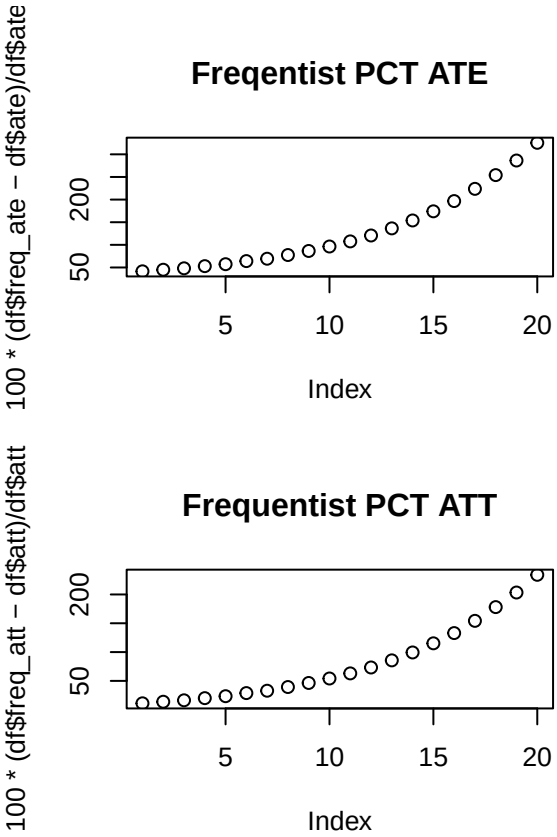







```
par(mfrow=c(2,2))
plot(100 * (df$xbcf_ate - df$ate) / df$ate , main = "XBCF PCT ATE")
plot(100 * (df$freq_ate - df$ate) / df$ate, main = "Freqentist PCT ATE")
plot(100 * (df$xbcf_att - df$att) / df$att, main = "XBCF PCT ATT")
plot(100 * (df$freq_att - df$att) / df$att, main = "Frequentist PCT ATT")
```


XBCF PCT ATE
100 * (df$xbcf_ate – df$ate)/df$ate
−20
−10
5
10
15
20
Index


Freqentist PCT ATE
100 * (df$freq_ate – df$ate)/df$ate
50
200
5
10
15
20
Index

XBCF PCT ATT
100 * (df$xbcf_att – df$att)/df$att
−20
−5
5
10
15
20
Index

Frequentist PCT ATT
100 * (df$freq_att – df$att)/df$att
50
200
5
10
15
20
Index