Dark Rate Estimation - Simulation Example

Nicole Bellert

2024-02-09

```
rm(list = ls())
```

Simulation of collusive and non-collusive behaviour and detection

Stigler ICC: delta > 1 - 1/nfirms
delta = c_value
C if delta >= 1 - 1/nfirms = ICC
same as delta <= 1/nfirms

Z is policy shock and happens after several time periods.

```
# Set random seed op <- options(digits.secs = 6) sim_seed <-
# as.numeric(Sys.time()) set.seed(sim_seed)

# Seed used for graphs and tables to get reproducible code
sim_seed <- 1673465635  # seed for enforcement
set.seed(sim_seed)
```

Set basic parameters

```
get_input <- function(id) {
   input <- tibble(
     sim_id = id,
     n = 99, # number of industries
     seed = sim_seed,
     periods = 4,
     share_no_z = 0.5, # share of time periods without policy shock
     sigma1 = 0.2,
     sigma2 = 0.4,
     sigma_re = 0.5
     )
}
```

```
get_basic_output <- function(input) {
    output <- input %>%
        mutate(D = 0, C = 0, ARE = 0, LATE = 0, E_D1D2Z = 0, E_D1D2 = 0, C_hat1 = 0,
            C_hat = 0, compliers = 0)
}
```

Build paneldata

```
build_panel <- function(ip) {
    nf <- 2:10  # number of firms in industry
    n_firms_in <- rep(nf, ip$n/length(nf))
    ind_id <- 1:ip$n
    t_in <- 1:ip$periods
    periods_no_z <- round(ip$periods * ip$share_no_z, 0)
    periods_z <- ip$periods - periods_no_z
    Z_in <- c(rep(0, periods_no_z), rep(1, periods_z))
    c_value_in <- runif(ip$n)
    d_value_in = runif(ip$n * ip$periods)
```

```r
    df <- tibble(sim_id = ip$sim_id, t = rep(t_in, each = ip$n), ind_id = rep(ind_id,
        length(t_in)), n_firms = rep(n_firms_in, length(t_in)), Z = rep(Z_in, each = ip$n),
        prob_c = 1/n_firms, prob_c_rep = 1/(2 * n_firms), c_value = rep(c_value_in,
            length(t_in)), prob_d = ifelse(Z == 0, ip$sigma1, ip$sigma2), prob_d_rep = ip$sigma_re,
        d_value = d_value_in, C = ifelse(t == 1, as.numeric(c_value <= prob_c), 0),
        D1 = 0, D2 = C * ifelse(t == 1, as.numeric(d_value <= prob_d), 0))
    df <- arrange(df, ind_id, t)
}
simulate_detection <- function(df) {
    for (j in 1:length(df$t)) {
        if (df$t[j] > 1) {
            df$D1[j] <- ifelse(df$D2[j - 1] == 1, 1, 0)  # 1 for detected yesterday.
            df$C[j] <- ifelse(df$D1[j] == 1, as.numeric(df$c_value[j] <= df$prob_c_rep[j]),
                as.numeric(df$c_value[j] <= df$prob_c[j]))
            df$D2[j] <- df$C[j] * ifelse(df$D1[j] == 1, as.numeric(df$d_value[j] <=
                df$prob_d_rep[j]), as.numeric(df$d_value[j] <= df$prob_d[j]))
        }
    }
    return(df)
}
add_complier <- function(df) {
    df <- df %>%
        group_by(ind_id) %>%
        mutate(complier1 = as.numeric((lead(D2 == 1) & (Z == 0)))) %>%
        mutate(complier1 = ifelse(is.na(complier1), 0, complier1)) %>%
        mutate(complier2 = as.numeric(lead(complier1 == 1))) %>%
        mutate(complier2 = ifelse(is.na(complier2), 0, complier2)) %>%
        mutate(complier3 = ifelse((complier1 + complier2) > 0, 1, 0)) %>%
        mutate(complier = max(complier3)) %>%
        mutate(D1D2 = D1 * D2)
}
```

Dark Rate Estimation

$ARE = E[D2 - D1D2] = E[D2 - D1D2|C2] * Pr(C2)$

LARE $= E[D2 - D1D2|complier]$

$\hat{E}(C) = ARE/LARE$

```r
estimate_output <- function(df, ip) {
    df_comp = df[df$complier == 1, ]
    op <- ip %>%
        mutate(D = mean(df$D2), C = mean(df$C), ARE = mean(df$D2 - df$D1D2), LATE = mean(df_comp$D2 -
            df_comp$D1D2), E_D1D2Z = mean(df$D1D2Z), E_D1D2 = mean(df$D1D2), C_hat1 = ARE/LATE,
            C_hat = ifelse(D > C_hat1, D, C_hat1), compliers = mean(df$complier))
    op
}
```

```r
n_sims <- 20
ip1 <- get_input(1)
output <- get_basic_output(ip1)

for (j in 1:n_sims) {
    ip <- get_input(j)
    pd <- build_panel(ip)
    pd <- simulate_detection(pd)
    pd <- add_complier(pd)
    op <- estimate_output(pd, ip)
    output[j, ] <- op
}
```

```r
output$avg_C <- cumsum(output$C)/output$sim_id
output$avg_D <- cumsum(output$D)/output$sim_id
output$avg_C_hat1 <- cumsum(output$C_hat1)/output$sim_id
output$avg_C_hat <- cumsum(output$C_hat)/output$sim_id
```

```r
ggplot(output) + geom_point(aes(x = sim_id, y = avg_C_hat, colour = "% Estimated cartels"),
    alpha = 1, size = 1.5) + geom_point(aes(x = sim_id, y = avg_C, colour = "% True cartels"),
    alpha = 1, size = 1.5) + geom_point(aes(x = sim_id, y = avg_D, colour = "% Detected cartels"),
    alpha = 1, size = 1.5) + scale_color_manual(name = "", values = c("green", "deepskyblue",
    "black")) + ylim(0, 0.3) + ylab("Cartel cases in % of industry population") +
    xlab("Multiple simulations, cumulative average of estimator")
```