Final Project

L. Gebrekristos

6/9/2022

Purpose

Using parametric g-formula, I estimated the effect of eviction on health by estimating how incidence of poor self-reported health is impacted when all experience a landlord-related forced move and when none experience a landlord-related forced move.

Needed Packages

```
library(dplyr)
library(tidyr)
library(gfoRmula)
library(ggplot2)
library(cowplot)
library(grid)
library(gridExtra)
library(stringr)
```

g-foRmula

```
dat <- evictph3


####### Monte Carlo sample size and bootstrap samples
ncores <- parallel::detectCores() - 1
nsamples <- 250
nsimul <- 2500
seed <- 1234

######## parameters id, time, # of time points
id <- "q1_1"
time_name <- "timept"
time_points <- 4

# time-varying covariates, baseline covariates, outcome var
timevarying <- c("income", "housingasst", "forcedmove_a", "incarfu")
timevaryingtypes <- c("normal", "binary", "binary", "binary")
```

```
baselinevars <- c("twrace", "male", "age", "edu", "incar")
outcome_name <- "poorhealth"

# timevarying models
covmodels <- c("income~
            lag1_forcedmove_a +
            lag1_income +
            lag1_incarfu +
            lag1_housingasst+
            as.factor(twrace) +
            male +
            age +
            incar +
            as.factor(edu) +
            as.factor(timept)",
    "housingasst ~
            lag1_forcedmove_a +
            lag1_income +
            lag1_housingasst +
            lag1_incarfu +
            as.factor(twrace) +
            male +
            age +
            incar +
            as.factor(edu) +
            as.factor(timept)",
    "forcedmove_a ~
            lag1_forcedmove_a +
            lag1_income +
            lag1_housingasst +
            lag1_incarfu +
            as.factor(twrace) +
            male +
            age +
            incar +
            as.factor(edu) +
            as.factor(timept)",
    "incarfu ~
            lag1_incarfu +
            lag1_forcedmove_a +
            lag1_housingasst +
            lag1_income +
            as.factor(twrace) +
            male +
            age +
            incar +
            as.factor(edu) +
            as.factor(timept)")

covparams <- list(covmodels = lapply(covmodels, function(x) as.formula(x)),
    control = c(NA, NA, list(maxit = 10000), list(maxit = 10000),
        NA))
```

```
# outcome model
ymodel <- poorhealth ~ forcedmove_a + housingasst + income +
    incarfu + lag1_forcedmove_a + lag1_income + lag1_housingasst +
    lag1_incarfu + twrace + male + age + incar + as.factor(edu) +
    as.factor(timept)

# define interventions: NEVER FORCED MOVE vs ALL FORCED
# MOVE
intvars = list(c("forcedmove_a"), c("forcedmove_a"))
int_descript <- c("Never FM", "Always FM")

gform_ph1 <- gformula(obs_data = dat, id = id, time_points = time_points,
    time_name = time_name, covnames = timevarying, outcome_name = outcome_name,
    outcome_type = "survival", covtypes = timevaryingtypes, covparams = covparams,
    ymodel = ymodel, intvars = intvars, interventions = list(list(c(static,
        rep(0, 4))), list(c(static, rep(1, 4)))), int_descript = int_descript,
    histories = c(lagged), histvars = list(c("income", "housingasst",
        "forcedmove_a", "incarfu")), basecovs = baselinevars,
    nsimul = nsimul, seed = seed, nsamples = nsamples, parallel = TRUE,
    ncores = ncores, ref_int = 1)
```

Table 1

```
cruderesults <- gform_ph1$result[10:12, c(2, 3, 4, 6, 7, 8, 10,
    11, 12, 14, 15)]

perc <- function(x) {
    if (is.numeric(x)) {
        ifelse(is.na(x), x, paste0(round(x * 100L, 1), "%"))
    } else x
}

inv <- function(x) {
    1/x
}

neg <- function(x) {
    -x
}

CI <- function(est, l, u) {
    paste0(est, " (", l, ", ", u, ")")
}
CI_V <- Vectorize(CI)

i0 <- cruderesults %>%
    filter(Interv. == 0) %>%
    mutate_at(vars(`RR lower 95% CI`, `RR upper 95% CI`, `Risk ratio`),
        funs(inv)) %>%
    mutate(lower = `RR upper 95% CI`) %>%
    mutate(`RR upper 95% CI` = `RR lower 95% CI`) %>%
```

```
    mutate(`RR lower 95% CI` = lower) %>%
    mutate_at(vars(`RD lower 95% CI`, `RD upper 95% CI`, `Risk difference`),
        funs(neg)) %>%
    select(!lower)

results <- rbind(i0, cruderesults[c(2:3), ])

results <- results %>%
    mutate_at(vars(`g-form risk`, `Risk lower 95% CI`, `Risk upper 95% CI`,
        `Risk difference`, `RD lower 95% CI`, `RD upper 95% CI`),
        funs(perc)) %>%
    mutate(RD95 = CI_V(`Risk difference`, `RD lower 95% CI`,
        `RD upper 95% CI`)) %>%
    mutate_at(vars(`RR lower 95% CI`, `RR upper 95% CI`, `Risk ratio`),
        funs(round(., 2))) %>%
    mutate_at(vars(`RR lower 95% CI`, `RR upper 95% CI`, `Risk ratio`),
        funs(round(., 2))) %>%
    mutate(RR95 = CI_V(`Risk ratio`, `RR lower 95% CI`, `RR upper 95% CI`)) %>%
    mutate(Risk95 = CI_V(`g-form risk`, `Risk lower 95% CI`,
        `Risk upper 95% CI`))


table1 <- data.frame(t(results))
table1 <- table1[c(14, 13, 12), ]
table1[c(2:3), 2] <- ""
rownames(table1) <- c("Risk (%, 95% CI)", "Risk Ratio (95% CI)",
    "Risk Difference (%, 95% CI)")
table1 <- table1 %>%
    mutate(`All Exposed` = X3) %>%
    mutate(`None Exposed` = X2) %>%
    mutate(Observed = X1) %>%
    select(`All Exposed`, Observed, `None Exposed`)


table1
```

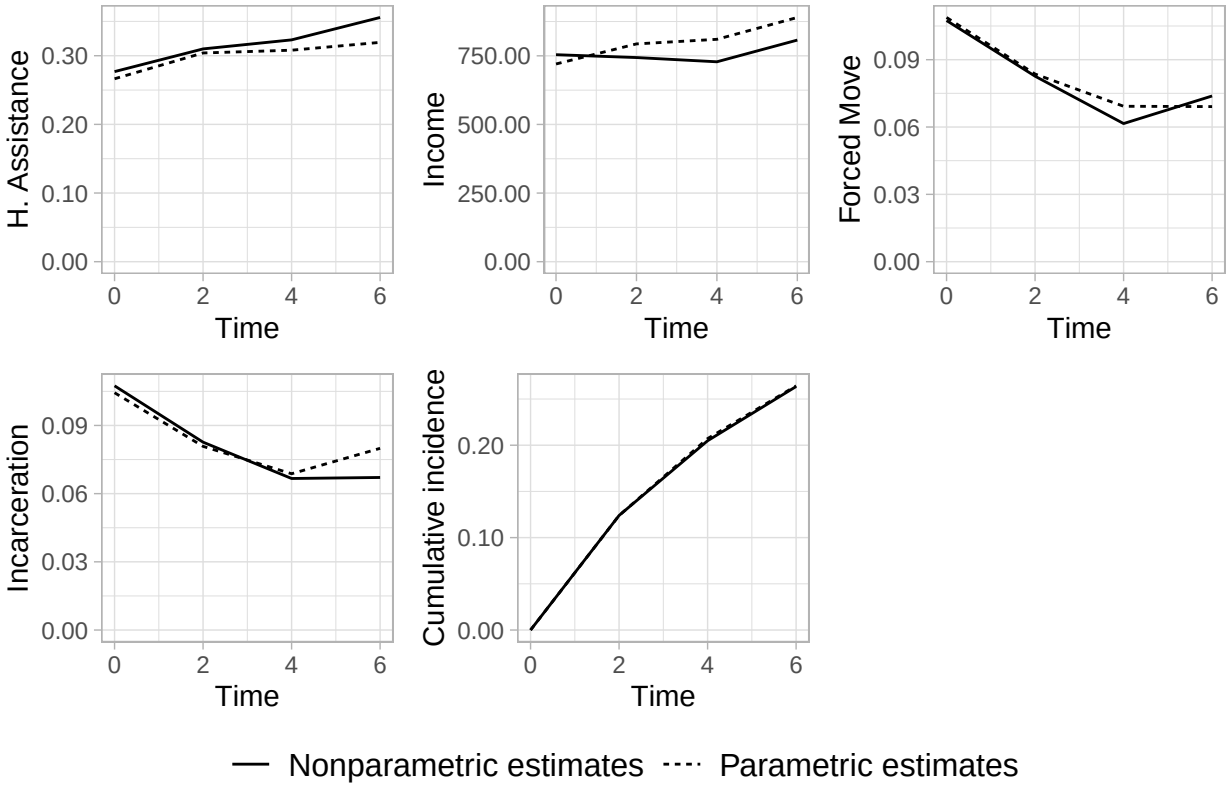
```
##                                      All Exposed             Observed
## Risk (%, 95% CI)            51.5% (22.3%, 77.7%) 26.4% (20.6%, 32.5%)
## Risk Ratio (95% CI)            2.13 (0.95, 3.63)    0.92 (0.82, 0.99)
## Risk Difference (%, 95% CI)   27.3% (-1.3%, 55%) -2.2% (-0.2%, -4.7%)
##                                     None Exposed
## Risk (%, 95% CI)            24.2% (18.9%, 29.7%)
## Risk Ratio (95% CI)
## Risk Difference (%, 95% CI)
```

Plots

```
# plots
cont_opts <- c("H. Assistance", "Income", "Forced Move", "Incarceration")
plot_cont <- function(var, num) {
```

```
    out <- data.frame(gform_ph1$dt_cov_plot[var])
    names(out) <- c("time", "cov", "legend")

    ggplot(out, aes(x = time * 2, y = cov, group = legend, lty = legend)) +
        geom_line() + theme_light() + ylab(cont_opts[num]) +
        xlab("Time") + scale_y_continuous(limits = c(0, max(out$cov)),
        labels = scales::number_format(accuracy = 0.01))
}

### function for outcome variable
plot_out <- function(var) {

    out <- data.frame(gform_ph1[var])
    names(out) <- c("time", "risk", "survival", "legend")

    ggplot(out, aes(x = time * 2, y = risk, group = legend, lty = legend)) +
        geom_line() + theme_light() + theme(legend.title = element_blank(),
        legend.position = "bottom", legend.text = element_text(size = 12)) +
        ylab("Cumulative incidence") + xlab("Time") + scale_y_continuous(limits = c(0,
        max(out$risk)), labels = scales::number_format(accuracy = 0.01)) +
        scale_linetype_discrete(labels = c("Nonparametric estimates",
            "Parametric estimates"))

}
ha <- plot_cont("housingasst", 1)
inc <- plot_cont("income", 2)
fa <- plot_cont("forcedmove_a", 3)
incar <- plot_cont("incarfu", 4)
out <- plot_out("dt_out_plot")
my_legend <- get_legend(out)
grid.arrange(arrangeGrob(ha + theme(legend.position = "none"),
    inc + theme(legend.position = "none"), fa + theme(legend.position = "none"),
    incar + theme(legend.position = "none"), out + theme(legend.position = "none"),
    nrow = 2), my_legend, nrow = 2, heights = c(10, 1), top = textGrob("Figure 1",
    gp = gpar(fontsize = 18, font = 1)))
```

Figure 1


H. Assistance
0.00
0.10
0.20
0.30
Income
0.00
250.00
500.00
750.00
Forced Move
0.00
0.03
0.06
0.09
Incarceration
0.00
0.03
0.06
0.09
Cumulative incidence
0.00
0.10
0.20
0
2
4
6
Time
Nonparametric estimates
Parametric estimates