```
#----------------------#
# SMART SUITE
# SPRING 2022
#
# CREATED BY:
#
# Giovanni M. Circo, Ph.D
# University of New Haven
# gcirco@newhaven.edu
#----------------------#

# Operation Impact
# Code-through example

# Code contains information on:
# 1:1, k:1 matching
# greedy and optimal matching
# propensity score estimation with a GAM
# propensity score weighting
# robust handling of standard errors

# you will need several packages for this example
# including:
# Matchit for matching
# sandwich for robust standard errors
# lmtest for applying robust SEs into a model

# be sure to install these first!
# install.packages("Matchit")
# install.packages("sandwich")
# install.packages("lmtest)

library(MatchIt)
library(sandwich)
library(lmtest)

set.seed(1)

# DATA SETUP
#----------------------#

# load the impact data

# file url
df <- url('https://www.dropbox.com/s/k1ndn3lvyty78x8/nyc_impact.csv?
raw=1')

# read into R
impact <- read.csv(df)
```

```
# variable descriptives:
# precinct 79 is the treatment precinct
# outcome variable is robberies in the 4th quarter
# `robbery.4`
head(impact)


# MATCHING
#----------------------#

# we are matching on:
# % male, %hispanic, %black, %poverty
# prior assault and robbery rates Q1 - Q3

# first, we set up a matching formula we can re-use
match_form <- impact ~
  total_male +
  total_hispan +
  total_black +
  total_poverty +
  assault.1 +
  assault.2 +
  assault.3 +
  robbery.1 +
  robbery.2 +
  robbery.3

# 1:1 greedy matching
# w/ with logit propensity score
match.1 <- matchit(match_form,
                   data = impact)

# 3:1 optimal matching
# w/ with logit propensity score
# allow up to 5 matches per control
# with control:treat ratio of 3:1
match.2 <- matchit(match_form,
                   data = impact,
                   method = "optimal",
                   min.controls = 1,
                   max.controls = 5,
                   ratio = 3)

# 3:1 optimal matching
# w/ with gam propensity score
# smooth terms over census variables
# NOTE: need to specify smooth terms
# with s()
match.3 <- matchit(impact ~
                     s(total_hispan) +
```

```
                      s(total_black) +
                      s(total_poverty) +
                      assault.1 +
                      assault.2 +
                      assault.3 +
                      robbery.1 +
                      robbery.2 +
                      robbery.3,
                    distance = "gam",
                    method = "optimal",
                    min.controls = 1,
                    max.controls = 5,
                    ratio = 3,
                    data = impact)

# get balance summaries
summary(match.1)
summary(match.2)
summary(match.3)

# plot std. mean differences
plot(summary(match.3), abs = F)

# plot covariate overlap
plot(match.3, "density")

# plot histogram of propensity scores
plot(match.3, "histogram")

# REGRESSION w/ MATCHING
#----------------------#

# get indexes of observations that are either
# in the treatment group, or were matched
wt <- match.3$weights > 0

# indexes of matched pairs
mpairs <- as.numeric(na.omit(match.3$subclass))

# fit model using only matched treatment/control group
# outcome is 4th quarter robberies `robbery.4`

fit.match <- lm(
  robbery.4 ~
    impact +
    total_hispan +
    total_black +
    total_poverty +
    assault.1 +
    assault.2 +
```

```
    assault.3 +
    robbery.1 +
    robbery.2 +
    robbery.3,
  data = impact[wt, ])

summary(fit.match)

# adjust standard errors
# cluster robust on the matched pairs
fit.match.CL <- vcovCL(fit.match, cluster = mpairs)
coeftest(fit.match, vcov. = fit.match.CL)

# REGRESSION w/ WEIGHTING
#----------------------#

# calculate inv prob. weights
impact_ipw <- impact

# get raw propensity scores
# by default, these are on the probability scale
impact_ipw$ps <- match.3$distance

# calculate the inverse probability weights
# 1/ps for treated units
# 1/(1-ps) for control units
impact_ipw$ipw <- ifelse(impact_ipw$impact == 1,
                         1/impact_ipw$ps,
                         1/(1-impact_ipw$ps))

fit.ipw <- lm(
  robbery.4 ~
    impact +
    total_hispan +
    total_black +
    total_poverty +
    assault.1 +
    assault.2 +
    assault.3 +
    robbery.1 +
    robbery.2 +
    robbery.3,
  data = impact_ipw,
  weights = ipw)

summary(fit.ipw)

# note: we need adjustments to the SEs
# can either use robust "HC3" or cluster bootstrap
# here cluster robust at the precinct level
```

```
# but bootstrap gives similar estimates as well

fit.ipw.cl  <- vcovCL(fit.ipw, cluster = impact$precinct)
coeftest(fit.ipw, vcov. = fit.ipw.cl)

# REGRESSION w/ STABALIZED WEIGHTING
#----------------------#

# alternative weights
# stabilized weights where
# if treat = 1: p/ps
# if treat = 0: 1- p/1-ps

# get marginal probability of treatment
p <- mean(impact$impact)

# calculate stabilized weights
impact_ipw$s_ipw <- ifelse(impact_ipw$impact == 1,
                           p/impact_ipw$ps,
                           1 - p/(1-impact_ipw$ps))

# fit model with stabilized weights
fit.ipw2 <- lm(
  robbery.4 ~
    impact +
    total_hispan +
    total_black +
    total_poverty +
    assault.1 +
    assault.2 +
    assault.3 +
    robbery.1 +
    robbery.2 +
    robbery.3,
  data = impact_ipw,
  weights = s_ipw)

# cluster robust errors
fit.ipw.cl2  <- vcovCL(fit.ipw2, cluster = impact$precinct)
coeftest(fit.ipw2, vcov. = fit.ipw.cl2)
```