GSA 2021 Poster Code

Peter Sun

October 22, 2021

1 Load Packages and Data

```r
library(tidyverse)
library(haven)
library(sjlabelled)
library(cobalt)
library(MatchThem)
library(mice)

# Avoid select clashes
select <- dplyr::select
recode <- dplyr::recode
summarize <- dplyr::summarize

# Import RAND 2010 HRS Fat File (hd10f5f)
rand.10 <- read_dta(Sys.getenv("HRS_2010_FAT"), col_select = c(

  # Identifiers
  hhid, pn,

  # Live alone = Number of people living in the household
  tot_hh_members = ma098,

  # Eyesight Rating (Section C, Wave 10 / Year 2010)
  # 1 = excellent, 2 = very good, 3 = good, 4 = fair, 5 = poor
  # 6 = leagally blind, # 8 = DK, # 9 = RF
  vision = mc095,

  # Religious Participation (Section B, Wave 10 / Year 2010)
  religious = mb082,

)) %>%
  rename(
  vision = mc095,
  religious = mb082,
  tot_hh_members = ma098)

# Import Region Early V8.2
r <- read_dta(Sys.getenv("HRS_REGION"), col_select = c(

  # Identifiers
```

```r
  hhid, pn,

  # HRS URBAN-RURAL CODE 2010 (BEALE 2013)
  beale2013_10

)) %>%
  rename(region = beale2013_10)

# Import Tracker Early 2018 V 2.0
t <- read_dta(Sys.getenv("HRS_TRACKER"), col_select = c(

  # Identifiers
  hhid, pn,

  # RESPONDENT WEIGHT FOR THE 2012 LEAVE BEHIND QNAIRE
  # See https://hrsonline.isr.umich.edu/sitedocs/wghtdoc.pdf
  nlbwgtr

)) %>%
  rename(lb_weight = nlbwgtr)

# Longitudinal RAND 1992_2018v1
rand <- read_dta(Sys.getenv("HRS_LONG"), col_select = c(

  # Identifiers
  hhid, pn,

  # Wave Information
  inw10, inw11,

  # Age at Exit Interview (Wave 10 / Year 2010)
  r10agey_e,

  # Gender (1 = male, 2 = female)
  ragender,

  # Race (1 = White/Caucasian, 2 = Black/African American, 3 = Other)
  raracem,

  # Marital Status (Wave 10 / Year 2010)
  # 2 = Married, spouse absent, 3 = partnered, 4 = separated,
  # 5 = divorced, 6 = separated/divorced, 7 = widowed,
  # 8 = never married
  r10mstat,

  # Educational Degree 1 = < HS, 2 = GED, 3 = HS grad,
  # 4 = some college, 5 = college and above
  raeduc,

  # Total Household Income (Wave 10 / Year 2010)
  h10iftot,

  # Self Report of Health (Wave 10 / Year 2010)
```

```r
    r10shlt,

    # ADL (Wave 10 / Year 2010)
    r10adla,

    # IADL (Wave 10 / Year 2010)
    r10iadlza,

    # Poverty (Wave 10 / Year 2010)
    h10inpov,

    # Chronic health conditions (Wave 10 / Year 2010)
    # (high bp, diabetes, cancer,
    # lung, heart, stroke, psych, arthritis)
    h1 = r10hibp, h2 = r10diab, h3 = r10cancr, h4 = r10lung, h5 = r10heart,
    h6 = r10strok, h7 = r10psych, h8 = r10arthr,

    # Depression (Wave 10 / Year 2010)
    r10cesd,

    # Nursing Home (Wave 10 / Year 2010)
    r10nhmliv,

    # Medicaid (Wave 10 / Year 2010)
    r10govmd,

    # Retirement
    r10sayret

)) %>%
rename(age = r10agey_e,
    gender = ragender,
    race = raracem,
    married = r10mstat,
    degree = raeduc,
    income = h10iftot,
    self_health = r10shlt,
    adl = r10adla,
    iadl = r10iadlza,
    poverty = h10inpov,
    h1 = r10hibp, h2 = r10diab, h3 = r10cancr,
    h4 = r10lung, h5 = r10heart, h6 = r10strok,
    h7 = r10psych, h8 = r10arthr,
    depression = r10cesd,
    nursing = r10nhmliv,
    medicaid = r10govmd,
    retired = r10sayret)

# Import Treatment, Outcome, and LBQ Survey Weights from
# the RAND 2012 HRS Core Fat File (h12f3a)
rand.12 <- read_dta(Sys.getenv("HRS_2012_FAT"), col_select =c(

    # Identifiers
```

```r
  hhid, pn,

  # HCBS Variables (NV051: attend congregate meal;
  # NV054: home delivered meals; NV057: transportation; NV060: case management
  # NV063: home services; NV068: caregiver services)
  # n6 = nv068, (including this is likely a mistake, see literature)
  n1 = nv051, n2 = nv054, n3 = nv057, n4 = nv060, n5 = nv063,

  # Social engagement (volitional)
  d1 = nlb001c, d2 = nlb001d, d3 = nlb001e, d4 = nlb001f, d5 = nlb001g, d6 = nlb001h,
  d7 = nlb001i, d8 = nlb001k, d9 = nlb001l, d10 = nlb001m, d11 = nlb001n, d12 = nlb001o,
  d13 = nlb001p, d14 = nlb001q, d15 = nlb001r, d16 = nlb001s, d17 = nlb001t,

  # Completed leave-behind questionnaire
  lb_comp = nlbcomp

)) %>%
  rename(n1 = nv051, n2 = nv054, n3 = nv057, n4 = nv060, n5 = nv063,
         d1 = nlb001c, d2 = nlb001d, d3 = nlb001e, d4 = nlb001f, d5 = nlb001g,
         d6 = nlb001h, d7 = nlb001i, d8 = nlb001k, d9 = nlb001l,
         d10 = nlb001m, d11 = nlb001n, d12 = nlb001o, d13 = nlb001p,
         d14 = nlb001q, d15 = nlb001r, d16 = nlb001s, d17 = nlb001t,
         lb_comp = nlbcomp)
```

2 Merge HRS Datasets and Clear Stata Formatting

```r
df <- rand.12 %>%
  left_join(t, by = c("hhid", "pn")) %>%
  left_join(r, by = c("hhid", "pn")) %>%
  left_join(rand, by = c("hhid", "pn")) %>%
  left_join(rand.10, by = c("hhid", "pn")) %>%
  haven::zap_formats() %>%
  sjlabelled::remove_all_labels() %>%
  as_tibble()
```

3 Recode Variables

```r
df.recoded <- df %>%

  # Recode marital status (0 = not married, 1 = married)
  # Marital status in wave 11 (2012), (1 = Married, 2 = Married, spouse absent,
  # 3 = partnered, 4 = separated,
  # 5 = divorced, 6 = separated/divorced,
  # 7 = widowed, 8 = never married, else NAs)
  mutate(married = ifelse(married == 1, 1,
    ifelse(married %in% 2:8, 0, NA))) %>%

  # Live alone (1 = live alone, 0 = does not live alone or at least two people
  # in the household, NA = NA)) (as a social isolation proxy)
  mutate(live_alone = ifelse(tot_hh_members == 0, 1,
    ifelse(tot_hh_members >= 1, 0, NA))) %>%

  # Recode education (1 = < HS & GED, 2 = HS grad,
  # 3 = some college & college and above)
  # Consider GED as less than HS based on
  # https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4112190/
  mutate(degree = dplyr::recode(degree,
    `1` = 1, `2` = 1, `3` = 2, `4` = 3, `5` = 3,
    .default = NA_real_)) %>%

  # Recode HCBS Variables
  # HCBS = 1 if there are any 1s (services)
  # Remove sub_food, because it asks about "you or other family members"
  mutate_at(vars(n1:n5),
    list(~dplyr::recode(., `1` = 1, `5` = 0,
    .default = NA_real_))) %>% # 0 = "No", 1 = "Yes"
  mutate(hcbs_na = rowSums(is.na(
    dplyr::select_at(.,vars(n1:n5))))) %>% # Count HCBS NAs
  mutate(hcbs_sum = rowSums(dplyr::select(., n1, n2, n3, n4, n5),
                            na.rm = TRUE)) %>% # Sum HCBSs
  mutate(hcbs = ifelse(
    hcbs_sum >= 1, 1, 0)) %>% # if at least one hcbs, then 1, otherwise 0
  mutate(hcbs = ifelse(
    hcbs_na == 5, NA, hcbs)) %>% # if 5 NAs, then NA, otherwise 0

  # Rural
  mutate(rural = recode(region, `1` = 0, `2` = 0, `3` = 1,
    .default = NA_real_)) %>%

  # Chronic Health Conditions
  mutate(chronic = rowSums(
    ifelse(
      select(., h1, h2, h3, h4, h5, h6, h7, h8) == 1, 1, 0
      ), na.rm = TRUE)) %>% # count 1s from h1-h8

  # Vision (1 = excellent, 2 = very good, 3 = good, 4 = fair, 5 = poor,
  # 6 = legally blind, 8 = don't know, 9 = refused)
  mutate(vision = recode(vision, `1` = 1, `2` = 1, `3` = 1, `4` = 0,
```

```r
    `5` = 0, `6` = 0, .default = NA_real_)) %>%

# Religious / How often attend religious service (1 = more than once a week,
# 2 = once a week, 3 = two or three times a week,
# 4 = one or more times a week, 5 = not at all, 8 = DK, 9 = RF)
mutate(religious = recode(religious, `1` = 1, `2` = 1, `3` = 1, `4` = 1,
  `5` = 0, .default = NA_real_)) %>%

# Social Engagement
mutate_at(vars(d1:d17), list(~dplyr::recode(., `1` = 1, `2` = 1, `3` = 1,
  `4` = 1, `5` = 0, `6` = 0, `7` = 0,
  .default = NA_real_))) %>% # recode
mutate(engagement_soc_sum = rowSums(
  dplyr::select_at(., vars(d1:d3, d5)), na.rm = TRUE)) %>%
mutate(engagement_soc_na = rowSums(
  is.na(dplyr::select_at(.,vars(d1:d3, d5))))) %>% # count NAs
mutate(engagement_soc_binary = ifelse(engagement_soc_sum >= 1, 1, 0)) %>%
mutate(engagement_soc_sum = ifelse(
  engagement_soc_na > 2, NA, engagement_soc_sum)) %>% # NA if > 2 NAs
mutate(engagement_soc_binary =
        ifelse(engagement_soc_na > 2, NA,
               engagement_soc_binary)) %>% # set to NA if more than 2 NAs

mutate(engagement_cog_sum = rowSums(dplyr::select_at(.,
  vars(d7:d11, d13:d15)), na.rm = TRUE)) %>%
mutate(engagement_cog_na = rowSums(is.na(dplyr::select_at(.,
  vars(d7:d11, d13:d15))))) %>% # count NAs
mutate(engagement_cog_binary = ifelse(
  engagement_cog_sum >= 1, 1, 0)) %>%
mutate(engagement_cog_sum = ifelse(
  engagement_cog_na > 2, NA,
  engagement_cog_sum)) %>% # set to NA if more than 2 NAs
mutate(engagement_cog_binary = ifelse(
  engagement_cog_na > 2, NA,
  engagement_cog_binary)) %>% # set to NA if more than 2 NAs

mutate(engagement_phy_sum = rowSums(dplyr::select_at(.,
  vars(d12, d16, d17)), na.rm = TRUE)) %>%
mutate(engagement_phy_na = rowSums(is.na(dplyr::select_at(.,
  vars(d12, d16, d17))))) %>% # count NAs
mutate(engagement_phy_binary = ifelse(
  engagement_phy_sum >= 1, 1, 0)) %>%
mutate(engagement_phy_sum = ifelse(engagement_phy_na > 2, NA,
  engagement_phy_sum)) %>% # set to NA if more than 2 NAs
mutate(engagement_phy_binary = ifelse(engagement_phy_na > 2, NA,
  engagement_phy_binary)) %>% # set to NA if more than 2 NAs

# Collapse ADL and IADL
# 0 = none, 1 = 1-2 ADLs, 2 = >=3 ADLs
mutate(adl = recode(adl, `0` = 0, `1` = 1, `2` = 1,
                  `3` = 2, `4` = 2, `5` = 2)) %>%
mutate(iadl = recode(iadl, `0` = 0, `1` = 1, `2` = 1,
                  `3` = 2, `4` = 2, `5`= 2)) %>%
```

```r
  # Follow Muramatsu for income and wealth transformations
  mutate(income = ifelse(income == 0, income, log(income))) %>%

  # Retirement
  # Original: 0 = not retired, 1 = completely retired, 2 = partly retired,
  # 3 = not working for pay or homemaker (change 3. question irrelevant to NA)
  mutate(retired = recode(retired, `0` = 0, `1` = 1, `2` = 2,
                          .default = NA_real_)) %>%

  # Convert Categorical Variables to Factors
  mutate(race = factor(race),
         degree = factor(degree),
         self_health = factor(self_health),
         adl = factor(adl),
         iadl = factor(iadl),
         retired = factor(retired))
```

3.1 Implement Sample Inclusion Criteria

```r
# Wave 11 (2012) Sample Size = 20554
nrow(df.recoded)
```

```
## [1] 20554
```

```r
table(df.recoded$inw10)
```

```
##
##     0     1
##   771 19783
```

```r
table(df.recoded$inw11)
```

```
##
##       1
## 20554
```

```r
# 1. >= Age 51 = 18,554
df2 <- df.recoded %>%
  filter(age >= 51)
nrow(df2)
```

```
## [1] 18554
```

```r
# 2. Not in nursing home = 18,293
df3 <- df2 %>%
  filter(nursing == 0)
nrow(df3)
```

```
## [1] 18293
```

```r
# 3. Completed the 2012 LB (i.e., potentially have outcome variables) = 6,764
df4 <- df3 %>%
  filter(lb_comp %in% c(1, 2, 4))
nrow(df4)
```

```
## [1] 6764
```

```r
# 4. Answered at least one HCBS question in Module 2 = 627
df5 <- df4 %>%
  filter(hcbs_na != 5)
nrow(df5)
```

```
## [1] 627
```

```r
# 5. No missing values in the dependent variables = 617
df6 <- df5 %>%
  filter(!is.na(engagement_soc_sum) &
           !is.na(engagement_cog_sum) &
           !is.na(engagement_phy_sum))
nrow(df6)
```

```
## [1] 617
```

4 Model Specifications

```
# Model Variables with Missing Data (n = 617)
df.final <- df6 %>%
  select(age, gender, race, married, degree, degree, income, self_health, adl, iadl,
         depression, live_alone, hcbs, rural,
         chronic, poverty, religious, vision,
         engagement_soc_sum, engagement_soc_binary,
         engagement_phy_sum, engagement_phy_binary, engagement_cog_sum,
         engagement_cog_binary, lb_weight, medicaid, retired)
nrow(df.final)
```

```
## [1] 617
```

```
# Model Variables Without Missing Data (n = 562)
df.final.cca <- df.final %>%
  drop_na()
nrow(df.final.cca)
```

```
## [1] 562
```

```
# % missing
1 - nrow(df.final.cca)/nrow(df.final)
```

```
## [1] 0.089141
```

5 Missing Data Analysis

```
df.final %>%
  dplyr::summarize(across(everything(), ~sum(is.na(.)))) %>%
  rowid_to_column() %>%
  pivot_longer(-rowid) %>%
  mutate(pct = value/nrow(df.final)) %>%
  arrange(desc(pct)) %>%
  mutate(pct = scales::percent(pct, accuracy = 0.1))
```

```
## # A tibble: 26 x 4
##    rowid name        value pct
##    <int> <chr>       <int> <chr>
## 1      1 retired        36 5.8%
## 2      1 depression     12 1.9%
## 3      1 medicaid        9 1.5%
## 4      1 race            6 1.0%
## 5      1 rural           5 0.8%
## 6      1 vision          2 0.3%
## 7      1 age             0 0.0%
## 8      1 gender          0 0.0%
## 9      1 married         0 0.0%
## 10     1 degree          0 0.0%
## # ... with 16 more rows
```

6 Multiple Imputation

```
# Multiple Imputation of Final Dataset
imp.m = 10
# df.final.imputes = mice(df.final, m = imp.m, maxit = 25, print = F)

# Load/Save Results
# save(list = c("df.final.imputes"), file = "data/df.final.imputes.Rdata")
load("data/df.final.imputes.Rdata")

# Inspect one of the complete data
df.final.imputed_data = complete(df.final.imputes, 5)
nrow(df.final.imputed_data)
```

```
## [1] 617
```

7 Propensity Score Weighting

```r
# Define Propensity Score Model Formula
f.match <- as.formula(
  "hcbs ~ age + gender + race + married + degree + live_alone +
  income + self_health + adl + iadl + rural + vision + poverty +
  chronic + religious + depression + lb_weight + medicaid + retired"
)

# Estimate Propensity Score Weights
# df.final.psw <- weightthem(
#   formula = f.match,
#   datasets = df.final.imputes,
#   approach = "within",
#   method = "gbm",
#   estimand = "ATT",
#   stop.method = "es.mean"
# )

# Load/Save Results
load("data/df.final.psw.Rdata")
# save(list = c("df.final.psw"), file = "data/df.final.psw.Rdata")

# Multiply the weights in each imputed dataset by the sampling weights
for(i in 1:imp.m) {
  df.final.psw$models[[i]]$weights <-
    df.final.psw$models[[i]]$weights * df.final.psw$models[[i]]$covs$lb_weight
}

# Love Plot Labels
love.plot.names <- c(
  age = "Age",
  gender_2 = "Female",
  race_1 = "Race - White/Caucasian",
  race_2 = "Race - Black/African American",
  race_3 = "Race - Other",
  married = "Married",
  degree_1 = "Education - Less than HS",
  degree_2 = "Education - HS Graduate",
  degree_3 = "Education - College and Above",
  live_alone = "Live Alone",
  income = "Income (Logged)",
  self_health_1 = "Self-Rated Health - Excellent",
  self_health_2 = "Self-Rated Health - Very Good",
  self_health_3 = "Self-Rated Health - Good",
  self_health_4 = "Self-Rated Health - Fair",
  self_health_5 = "Self-Rated Health - Poor",
  adl_0 = "ADLs - None",
  adl_1 = "ADLs - 1-2 ADLs",
  adl_2 = "ADLs - 3+ ADLs",
  iadl_0 = "IADLs - None",
  iadl_1 = "IADLs - 1-2 IADLs",
  iadl_2 = "IADLs - 3+ IADLs",
  rural = "Rural",
```

```
  vision = "Vision Score",
  poverty = "Poverty",
  chronic = "Chronic Conditions",
  religious = "Religious",
  depression = "Depression",
  lb_weight = "Survey Weight",
  medicaid = "Medicaid Coverage",
  retired_0 = "Retired - Not Retired",
  retired_1 = "Retired - Completely Retired",
  retired_2 = "Retired - Partly Retired"
  )

# Check Balance
lp <- love.plot(df.final.psw,
  binary = "std",
  stars = "std",
  drop.distance = TRUE,
  var.names = love.plot.names,
  shapes = c("circle filled", "circle"),
  threshold = c(m = .25),
  position = "top",
  title = NULL)
lp
```


Range across imputations
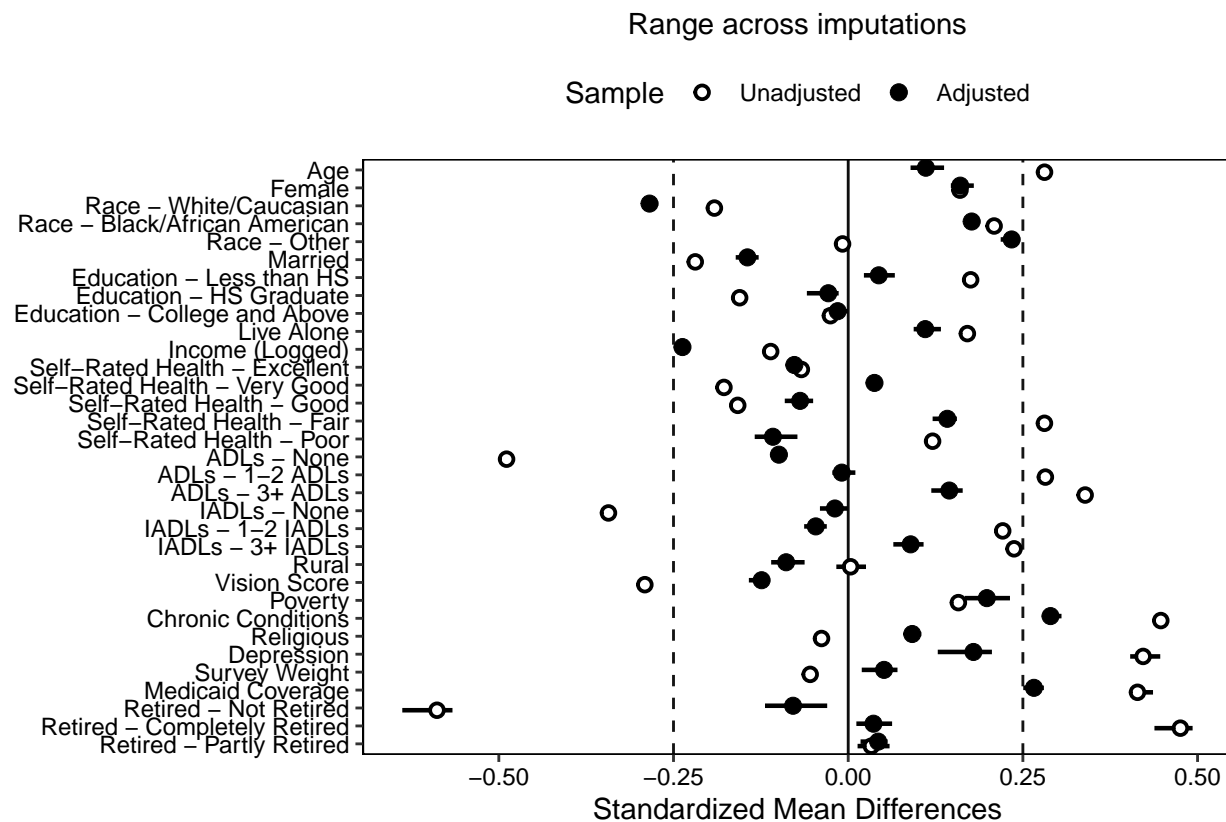
8   Genetic Matching

```
# # Genetic matching
# cl <- parallel::makeCluster(8)
# m1.gen <- matchthem(formula = f.match,
#            datasets = df.final.imputes,
#            approach = "within",
#            method = "genetic",
#            distance = "logit",
#            discard = "none",
#            estimand = "ATT",
#            pop.size = 1000,
#            cluster = cl)

# Load/Save Results
load("data/m1.gen.Rdata")
# save(list = c("m1.gen"), file = "data/m1.gen.Rdata")

# Check Balance
lp2 <- love.plot(m1.gen,
  binary = "std",
  stars = "std",
  var.names = love.plot.names,
  drop.distance = TRUE,
  shapes = c("circle filled", "circle"),
  threshold = c(m = .25),
  position = "top",
  title = "Covariate Balance After Genetic Matching")
lp2
```
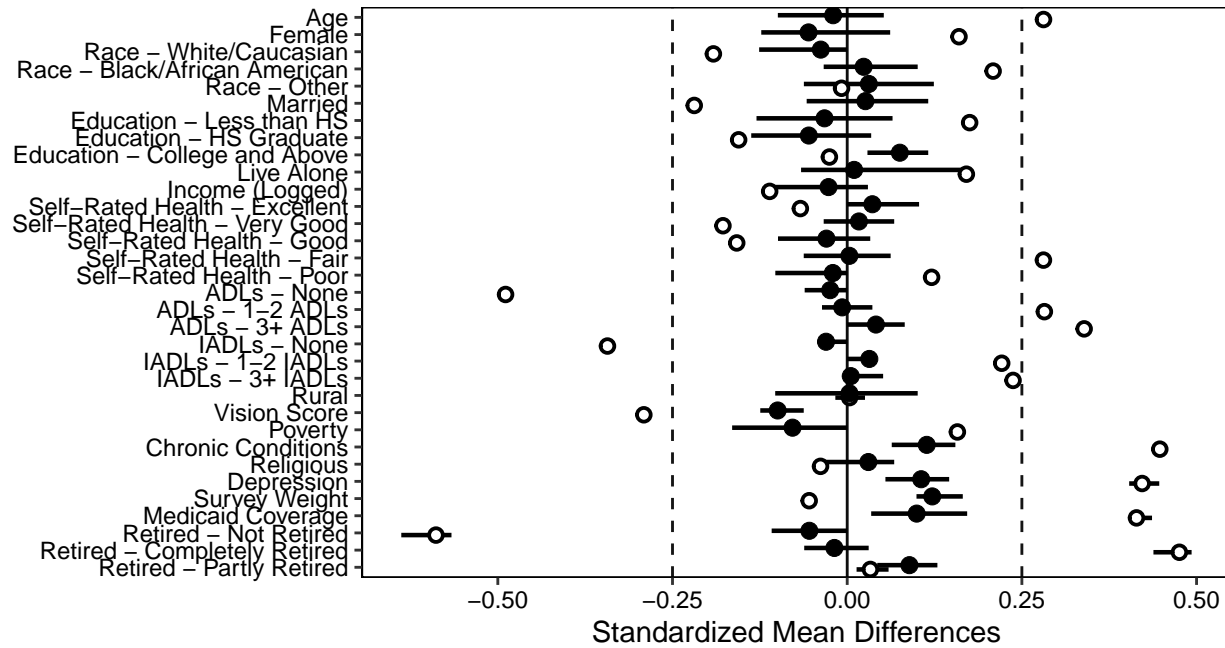
Covariate Balance After Genetic Matching
Range across imputations

```
ggsave("genetic.balance.png", dpi = 600, plot = lp2,
       width = 10, height = 5, units = "in")
```

9 Outcome Analysis with Genetic Matching

```
m1.gen.lm <- with(
  data = m1.gen,
  exp = lm(engagement_soc_sum ~ hcbs + age + gender +
           race + married + degree + live_alone +
           income + self_health + adl + iadl +
           rural + vision + poverty + chronic +
           religious + depression + medicaid + retired)
)
m1.gen.lm.pooled <- pool(m1.gen.lm)
summary(m1.gen.lm.pooled)
```

```
##            term     estimate   std.error    statistic       df   p.value
## 1   (Intercept)  0.200941293 0.756002735  0.26579440 72.24787 0.79115457
## 2          hcbs  0.277874483 0.129449404  2.14658758 50.81470 0.03662256
## 3           age  0.007047488 0.009148771  0.77032071 70.60896 0.44367946
## 4        gender  0.167167501 0.148405270  1.12642564 70.42872 0.26380894
## 5         race2  0.188630182 0.188794054  0.99913200 97.48868 0.32020427
## 6         race3 -0.003699872 0.352542215 -0.01049484 45.38102 0.99167249
## 7       married  0.065121984 0.142507935  0.45697093 84.32822 0.64886694
## 8       degree2  0.099279218 0.206532919  0.48069440 43.96240 0.63311735
## 9       degree3  0.168959215 0.177308684  0.95290998 45.31365 0.34569136
## 10   live_alone -0.167971743 0.177521253 -0.94620638 54.32122 0.34823348
## 11       income -0.224790683 0.203889106 -1.10251444 77.45653 0.27365198
## 12 self_health2 -0.077489727 0.297334845 -0.26061435 51.39937 0.79543122
## 13 self_health3 -0.011572981 0.304812333 -0.03796756 51.19293 0.96986122
## 14 self_health4 -0.062859283 0.315801614 -0.19904674 44.10809 0.84314089
## 15 self_health5 -0.260968785 0.365720861 -0.71357369 58.91065 0.47830835
## 16         adl1  0.014232696 0.197106434  0.07220818 72.53106 0.94263499
## 17         adl2  0.330575340 0.315931940  1.04634986 68.19087 0.29909438
## 18        iadl1 -0.397254912 0.243091213 -1.63418047 80.62804 0.10611983
## 19        iadl2 -0.756222443 0.412022443 -1.83539139 62.68003 0.07119282
## 20        rural -0.005474225 0.170568692 -0.03209396 47.24360 0.97453233
## 21       vision -0.094040037 0.165283260 -0.56896286 78.89341 0.57099758
## 22      poverty -0.275951176 0.209847811 -1.31500621 65.21158 0.19311337
## 23      chronic -0.059699739 0.050343523 -1.18584746 54.90306 0.24078941
## 24    religious -0.120447449 0.183018764 -0.65811530 70.82804 0.51259751
## 25   depression  0.017358731 0.042688645  0.40663579 35.30153 0.68673039
## 26     medicaid  0.020773358 0.213003472  0.09752591 64.97022 0.92260900
## 27      retired1 -0.393753504 0.223538042 -1.76146083 59.69176 0.08328139
## 28      retired2 -0.034147105 0.281793795 -0.12117763 48.80763 0.90404785
```

10 Outcome Analysis with Propensity Score Weighting

```
# Outcome Model
m1 <- with(
  data = df.final.psw,
  exp = lm(engagement_soc_sum ~ hcbs + age + gender +
           race + married + degree + live_alone +
           income + self_health + adl + iadl +
           rural + vision + poverty + chronic +
           religious + depression + medicaid + retired)
)
m1.pooled <- pool(m1)
summary(m1.pooled)
```

```
##             term     estimate   std.error  statistic        df         p.value
## 1    (Intercept) -0.191090738 0.286337402 -0.6673621 540.1524 0.504825851340
## 2           hcbs  0.114164161 0.053005504  2.1538171 579.5736 0.031666161757
## 3            age  0.006871696 0.003737382  1.8386392 554.9440 0.066502449773
## 4         gender  0.160654822 0.056362007  2.8504099 567.6997 0.004524941088
## 5          race2  0.117895739 0.093211646  1.2648177 547.8560 0.206474439614
## 6          race3  0.196608130 0.122728883  1.6019712 557.3454 0.109728565009
## 7        married  0.087139326 0.062747271  1.3887349 538.0836 0.165487914210
## 8        degree2  0.254835887 0.079173637  3.2186962 528.5174 0.001366780201
## 9        degree3  0.181950647 0.070120285  2.5948361 568.6460 0.009707785422
## 10    live_alone -0.158575941 0.066483216 -2.3852026 577.4786 0.017391247937
## 11        income -0.118041473 0.083877143 -1.4073139 551.4793 0.159897670585
## 12  self_health2 -0.171411283 0.112299675 -1.5263738 557.3840 0.127484013496
## 13  self_health3  0.060731993 0.113485629  0.5351514 560.7890 0.592757307793
## 14  self_health4  0.117065584 0.120195676  0.9739584 529.8074 0.330521678131
## 15  self_health5 -0.111199722 0.141830411 -0.7840330 557.0532 0.433353995568
## 16          adl1 -0.125926366 0.083001273 -1.5171619 571.8456 0.129778424247
## 17          adl2  0.310930107 0.149040402  2.0862135 537.0219 0.037430544893
## 18         iadl1 -0.218113464 0.101158858 -2.1561479 514.0584 0.031535318132
## 19         iadl2 -0.890753133 0.180629622 -4.9313790 501.5708 0.000001111999
## 20          rural  0.024303752 0.064150185  0.3788571 570.5708 0.704934933840
## 21         vision -0.088660593 0.067500765 -1.3134754 573.1667 0.189548337497
## 22        poverty -0.176213836 0.096654110 -1.8231386 581.2842 0.068795873358
## 23        chronic -0.040467937 0.020254770 -1.9979461 569.2683 0.046197791256
## 24      religious  0.072957938 0.067420152  1.0821384 538.9169 0.279674979707
## 25     depression  0.014674202 0.015898486  0.9229937 361.7856 0.356625643186
## 26       medicaid  0.171232053 0.102407938  1.6720584 522.7100 0.095111245649
## 27       retired1 -0.199513190 0.083047457 -2.4023997 530.1625 0.016630740426
## 28       retired2  0.063536454 0.101750673  0.6244328 297.6015 0.532821817826
```