# Class exercise week 9 - Designing weights

Peter Lugtig

16 oktober, 2023

## Question 1: diagnostics

Consider that you succeeded to draw a relatively large and perfectly random sample from your target population. This means that with fully observed data (the 1517 cases in the file "unit non response information.RDS"), the sample would be representative for the population (no worries about coverage and sampling errors for now). However, not every respondent that was addressed was willing to fill in your questionnaire, so unfortunately there is 'unit non-response'. This possibly creates bias. Luckily, we still have some basic information about the non-responders, so it is possible to weight back to the full sample (representing the target population).

The first part of the exercise is about checking for the potential bias (a). In the second part (b), you are asked to answer some research-questions correcting for this possible bias. a. Open the data file "unit non response information.RDS". In this file you find all available information about both responders and non-responders. Examine if there are any large differences between respondents and non-respondents with regard to region, gender, education, race, and age (x). In other words: are there strong predictor of response (r)? For this youcan run bivariate relationships, or predict the variable "response". The following variables are available:

id - Respondent id
region - 1= Northeast USA, 2= Southern US, 3= West sex - 1 = male, 2= female educ - higher number mean higher levels race - 1 = white, 2 = black, 3 = other (hispanic) age - Age in years response - 0 = nonresponse, 1 = response

## Question 2: Creating a weight

Two variables were selected for the subsequent weighting exercises: sex and race. Consider the sample to be unbiased with respect to all other variables. Compute the Nonresponse weight for the variable sex and race. Compute two weights (for each variable one), and then multiply both weights to achieve one nonresponse weights in your dataset. The easiest way to do this is to run a glm() with a logit link function, and then add the fitted.values (these are the propensity scores) to your dataset. The inverse of these propensity scores is your weight.

We now want to add the weights not to the dataset that contains respondents and nonrespondents, but only to the dataset with respondents. The code below loads the datafile "responders_data.RDS", recodes missing values and then adds the weights you created in the previous question to the respondent dataset. Run the code and make sure you have an additional weight variable in your dataset.

```
# now merge with substantive data
respdata <- readRDS("responders_data.RDS") # read data in as data frame
respdata$income[respdata$income==999999] <- NA

finaldata <-merge(respdata,data)
```

## Question 3: explore other ways to create weights.

There are other ways to create weights. We discussed the following four methods in the lecture:

1. Using poststratification (use the lecture slides about nonresponse). The basic idea is here that you compute a weight for every cell in the multi-way crosstable of your weight-variables.

2. Using linear weighting using population totals
3. Using a propensity based model using the population distribution
4. UsingRaking

The general syntax for these methods look like this: Poststratification:
- postStratify(design= svy.unweighted, strata = sexrace, population=sexrace.dist)

Linear weighting:
- calibrate(design=svy.unweighted,formula=~sex+race, population=c(sex.dist,race.dist), calfun=c("linear"))

Propensity score weighting (using population totals): - calibrate(design=svy.unweighted,formula=~sex+race, population=c(sex.dist,race.dist), calfun=c("logit"))
Raking:
- rake(design=svy.unweighted,sample.margins=list(sex,race), population.margins=list(sex.dist,race.dist))

Explore each of the weighting methods. Adjust the code below so you also include the variable "region". You may also add other variables, just to try out how these methods work.

```
# create sexrace crosstable
svy.unweighted <-svydesign(ids=~id, data=respdata)
```

```
## Warning in svydesign.default(ids = ~id, data = respdata): No weights or
## probabilities supplied, assuming equal probability
```

```
sexrace<- as.data.frame(table(respdata$sex,respdata$race))
sexrace.dist <-xtabs(~sex+race, data=data)

poststratdesign <- postStratify(design= svy.unweighted, strata =~sex+race, population=sexrace.dist)

# for Calibration (linear weighting)
sex.dist <- as.data.frame(table(data$sex))
colnames(sex.dist) <- c("sex", "Freq")
race.dist <- as.data.frame(table(data$race))
colnames(race.dist) <- c("race", "Freq")
table(data$sex,data$race)
pop.totals<-c('(Intercept)'=545+719,sexFemale=719+133+29,raceBlack=133+71,raceOther=20+29)
# you need to get the totals as numbers.

linearweightdesign <- calibrate(design=svy.unweighted,formula=~sex+race,
                                population=pop.totals, calfun=c("linear")) #linear weighting
propweightdesign  <- calibrate(design=svy.unweighted,formula=~sex+race,
                               population=pop.totals, calfun=c("logit"), bounds=c(0.3,3))
#inverse probability (propensity)
# a note about the bounds here -> they are needed to make sure the model converges.
# I trmmed at 3, but you can choose other values
# remember bias/variance trade-off?

rakeddesign  <- rake(design=svy.unweighted,sample.margins=list(~sex,~race),
            population.margins=list(sex.dist,race.dist))
```

## Question 4: does it matter?.

Compute the mean and variance in 'income' for the respondents without weights and then using the (three) weights that you created in the previous question. Do you find any differences between the weighted and

unweighted estimates? And does the weighting method make a difference? Do you think that variance inflation due to weighting is a problem here?

## Question 5 (optional)

How large is the difference in income between men and women under all weighting scenarios?

```
t.test(finaldata$income~ finaldata$sex)
svyttest(income~sex, design=poststratdesign,family=gaussian)
# etc.
```

## Question 6 (optional if you want to practice more):

To get a more real-life situation, I looked up some employment statistics in the USA using population data. The variable 'employm' indicates for respondents whether they have a job (1) or not (0). The population distribution for employment is as follows:

Employed: 48,2% Not employed: 51.8%

Use raking to correct for nonresponse using this additional variable. Use employm + the variables you used earlier in this exercise. Inspect the effects of using the additional variables on the mean and variance of 'income'.

`Example code:`

rake(design = svy.unweighted,(sample.margins = list(~sex, ~race, ~employm),(population.margins = list(sex.dist, race.dist, employ.dist))

– End of document –