# Stat472/572 Sampling: Design and Analysis


Instructor: Yan Lu

## Chapter 7: Complex Surveys

# Topic Overviews

- Complex surveys
- Survey plots
- Sampling and experimental design

# Building blocks for surveys

- Cluster sampling with replacement
- Cluster sampling without replacement
- Stratification
- Ratio estimation
- Weights
- Computer intensive methods

# Cluster sampling with replacement

- Select a sample of $n$ clusters with replacement
  —-suppose cluster $i$ is selected with probability $\psi_i$
- Estimate the total for cluster $i$ by an unbiased estimate $\hat{t}_i$
- Estimator for population total is $\hat{t}_\psi = \frac{1}{n}\sum_{i=1}^{N}\sum_{j=1}^{Q_i}\frac{\hat{t}_{ij}}{\psi_i}$
- $\hat{V}(\hat{t}_\psi) = \frac{1}{n}\sum_{i=1}^{N}\sum_{j=1}^{Q_i}\left(\frac{\hat{t}_{ij}}{\psi_i} - \hat{t}_\psi\right)^2/(n-1)$

# Cluster sampling without replacement

- Select a sample of $n$ clusters without replacement
- Suppose cluster $i$ is selected in the sample with probability $\pi_i$
- Estimate the total for cluster $i$ by an unbiased estimate $\hat{t}_i$
- Use the Horvitz-Thompson estimate of the population total

$$\hat{t}_{\mathrm{HT}} = \sum_{i \in S} \hat{t}_i / \pi_i$$

- $\hat{V}(\hat{t}_{\mathrm{HT}}) =$

$$\left(1 - \frac{n}{N}\right) \frac{n}{n-1} \sum_{i \in S} \left( \frac{\hat{t}_i}{\pi_i} - \frac{1}{n} \sum_{j \in S} \frac{\hat{t}_j}{\pi_j} \right)^2 + \sum_{i \in S} \frac{\hat{V}(\hat{t}_i)}{\pi_i}$$

# Stratification

- Estimate the strata totals by $\hat{t}_1, \hat{t}_2, \cdots, \hat{t}_H$
- The estimated variances for the strata totals are $\hat{V}(\hat{t}_1), \hat{V}(\hat{t}_2), \cdots, \hat{V}(\hat{t}_H)$
- The estimator of the population total is $\hat{t} = \sum_{i=1}^{H} \hat{t}_i$
- The estimator of the variance is $\hat{V}(\hat{t}) = \sum_{i=1}^{H} \hat{V}(\hat{t}_i)$

## Ratio estimation

- Let $\hat{t}_x$ and $\hat{t}_y$ be estimators of $t_x$ and $t_y$, respectively
- The ratio is estimated by $\hat{B} = \hat{t}_y / \hat{t}_x$
- The ratio estimator of the population total is $\hat{t}_{yr} = \hat{B} t_x$
- The estimated variance is $t_x^2 \hat{V}(\hat{B})$
- We often use ratio estimators for means, letting the auxiliary variable $x$ be an indicator (1 or 0) variable for whether or not unit $i$ is in the sample
  —-Here, $\hat{t}_x$ is an estimator of the population size and the ratio is the estimator of the population total divided by the estimated population size

# Stratification and Clustering

- $H$ strata, within stratum $h$, select $n_h$ psu's
- stratification: $\hat{t} = \sum_{h=1}^{H} \hat{t}_h$
- clustering part: $\hat{t}_h = \sum_{i \in S_h} \frac{\hat{t}_{hi}}{\pi_{hi}}$
- final total $\hat{t} = \sum_{h=1}^{H} \sum_{j \in S_h} \frac{\hat{t}_{hi}}{\pi_{hi}}$
- $\hat{V}(\hat{t}) = \sum_{h=1}^{H} \hat{V}(\hat{t}_h)$

# Any design

Let $w_i$ be weight for unit $i$

- $\hat{t} = \sum_{i \in S} w_i y_i$
- $\hat{\bar{y}} = \sum_{i \in S} w_i y_i / \sum_{i \in S} w_i$

  ——$\sum_{i \in S} w_i$ is used to estimate the population size, generally, not equal to true population size, except SRS, but should be close.

**Example 7.1:** Malaria in The Gambia

Malaria is a serious health problem in The Gambia. Malaria morbidity can be reduced by using bed nets that are impregnated with insecticide, but this is only effective if the bed nets are in widespread use. In 1991, a nationwide survey was designed to estimate the prevalence of bed net use in rural areas. The survey is described and results are reported in D'Alessandro et al.(1994).

| Stage | Stratification | Sampling units |
|---|---|---|
| 1 | Region | District |
| 2 | PHC/non-PHC | Village |
| 3 | | Compound |

- The sampling frame consists: all rural villages of fewer than 3000 people in The Gambia.
- The villages were stratified by three geographic regions (eastern, central, and western).
- In each region (eastern, central, western), four districts were chosen with probability proportional to the district population (used the 1983 census).
- The district were stratified by whether the village had a public health clinic (PHC) or not. Four villages were chosen, with probability proportional to the 1983 census population: two PHC villages and two non-PHC villages.
- Six compounds were chosen randomly from each village, and a researcher recorded the number of beds and nets, along with other information, for each compound.

## Simplicity in Survey Design

- A simpler design giving the same amount of information per dollar spent is almost always to be preferred to a more complicated design
  —-often easier to administer and analyze, and data from the simple designed survey are less likely to be analyzed incorrectly by subsequent analysts.
- Choose a more complex design if it is really more efficient and practical.

# Sampling Weights

- Stratified random sampling

$$\hat{t}_{\text{str}} = \sum_{h=1}^{H} \sum_{j \in S_h} w_{hj} y_{hj}$$

$\pi_{hj} = n_h / N_h$: probability of selecting the $j$th unit in the $h$th stratum to be in the sample
$w_{hj} = N_h / n_h$

$$\bar{y}_{\text{str}} = \frac{\sum_{h=1}^{H} \sum_{j \in S_h} w_{hj} y_{hj}}{\sum_{h=1}^{H} \sum_{j \in S_h} w_{hj}}$$

- Cluster sampling with equal probabilities

$$w_{ij} = \frac{NM_i}{nm_i} = \frac{1}{\text{probability that the jth ssu in the ith psu is in the sample}}.$$

$$\hat{t} = \sum_{i \in S} \sum_{j \in S_i} w_{ij} y_{ij}$$

$$\bar{y} = \frac{\hat{t}}{\sum_{i \in S} \sum_{j \in S_i} w_{ij}}$$

- Cluster sampling with unequal probabilities
  Let $w_i$ be the weight that the $i$th psu is in the sample
  and $w_{j|i}$ be the weight that the $j$th ssu is in the sample given that the $i$th psu is in the sample
  the sampling weights are

$$w_{ij} = w_i \times w_{j|i}$$

- Three-stage cluster sampling
  $w_p$: weight for the psu
  $w_{s|p}$: weight for the ssu
  $w_{t|s,p}$: weight associated with the tsu (tertiary sampling unit)
  the overall sampling weight for an observation unit is

$$w = w_p \times w_{s|p} \times w_{t|s,p}$$

- Any design
  $w_i$: weight for unit $i$

$$\hat{t} = \sum_{i \in S} w_i y_i$$

$$\hat{\bar{y}} = \sum_{i \in S} w_i y_i / \sum_{i \in S} w_i$$

Comments:

- All the information needed to construct point estimators is contained in the sampling weights
- However, the sampling weights give no information on how to find standard errors of the estimators, and thus knowing the sampling weights alone will not allow you to do inferential statistics
- Variances of estimators depend on the probabilities that any pair of observation units is selected to be in the sample, and requires more knowledge of the sampling design than given by weights alone

# Estimating a Distribution Function

Review: For a discrete random variable $X$ with possible values $x_1, x_2, \cdots, x_n$, a probability mass function is a function such that

$$f(x_i) \geq 0, \quad \sum_{i=1}^{n} f(x_i) = 1, \quad f(x_i) = P(X = x_i)$$

Cumulative distribution function is defined as

$$F(X) = P(X \leq x) = \sum_{x_i \leq x} f(x_i)$$

Continuous case

$$F(x) = P(X \leq x) = \int_{-\infty}^{x} f(x)dx$$

$$u = \int_{-\infty}^{+\infty} xdF(x) = \int_{-\infty}^{+\infty} xf(x)dx$$

For sampling, suppose the values for the entire population of $N$ units are known

- Probability mass function (pmf) of the finite population is

$$f(y) = \frac{\text{number of units whose value is y}}{N}$$

- Cumulative distribution function (cdf) of the finite population is

$$F(y) = \frac{\text{number of units with value} \leq y}{N} = \sum_{x \leq y} f(x)$$

- Finite population mean $\bar{y}_U = \sum_{\text{values of y in population}} yf(y)$

- Population median: any value $m$ with

$$F(m) \geq 1/2 \quad \text{and} \quad P(Y \geq m) \geq 1/2$$

- In general, $x$ is a 100rth percentile if

$$F(x) \geq r \quad \text{and} \quad P(Y \geq x) \geq 1 - r$$

- Population variance

$$\begin{aligned} S^2 &= \frac{1}{N-1} \sum_{i=1}^{N} (y_i - \bar{y}_U)^2 \\ &= \frac{N}{N-1} \sum_{y} f(y) \left[ y - \sum_{y} y f(y) \right]^2 \\ &= \frac{N}{N-1} \left[ \sum_{y} y^2 f(y) - \left( \sum_{y} y f(y) \right)^2 \right] \end{aligned}$$

## Estimators

- Empirical probability mass function (epmf)

$$\hat{f}(y) = \frac{\sum_{i \in S, y_i = y} w_i}{\sum_{i \in S} w_i}$$

- Empirical distribution function (ecdf)

$$\hat{F}(y) = \sum_{x \leq y} \hat{f}(x)$$

or

$$\hat{F}(y) = \frac{\sum_{i \in S, y_i \leq y} w_i}{\sum_{i \in S} w_i}$$

- Estimate the median by $\hat{F}(y) \approx 1/2$
- Estimate $\bar{y}_U$ by

$$\hat{\bar{y}} = \sum y \hat{f}(y) = \frac{\sum_{i \in S} y_i w_i}{\sum_{i \in S} w_i}$$

- Estimate population variance by

$$s^2 = \frac{N}{N-1} \left[ \sum_y y^2 \hat{f}(y) - \left( \sum_y y \hat{f}(y) \right)^2 \right]$$

# Plotting Data from Complex Survey

- plots commonly used for SRS can mislead when applied to raw data from non-self-weighting samples
- clustering causes numerous difficulties in plotting data from a complex survey, because the clustering structure as well as possible unequal weighting must be displayed in the graphs
- the problems are compounded because data sets from surveys are often very large and involve several layers of clustering
- plot data both with and without weights to see the effect of the weights
- plot data separately for each stratum and for each psu, if possible, to examine variability in the responses
- incorporating the weights into the graphics

**Example:**
Consider an artificial population of 1000 men and 1000 women stored in the file `htpop.dat`. Each individual's height is measured to the nearest centimeter (cm).
Two samples are drawn from this population:

- `htsrs.dat`: A simple random sample (SRS) of size 200.
- `htstrat.dat`: A stratified random sample consisting of 160 women and 40 men.

```r
# Empirical pmf for stratified sample of heights
# define sampling weight
library(survey)

## Loading required package: grid
## Loading required package: Matrix
## Loading required package: survival
##
## Attaching package: 'survey'
## The following object is masked from 'package:graphics':
##
##     dotchart

library(SDAResources)
library(tibble)
htstrat$sampwt <- 1000/sum(htstrat$gender=="F")
htstrat$sampwt[htstrat$gender=="M"] <-
  1000/sum(htstrat$gender=="M")
head(htstrat)

## # A tibble: 6 x 4
##      rn height gender sampwt
##   <dbl>  <dbl> <chr>   <dbl>
```

```
## 1   201    166 F        6.25
## 2   965    163 F        6.25
## 3   490    166 F        6.25
## 4   249    155 F        6.25
## 5   260    154 F        6.25
## 6   324    160 F        6.25
```

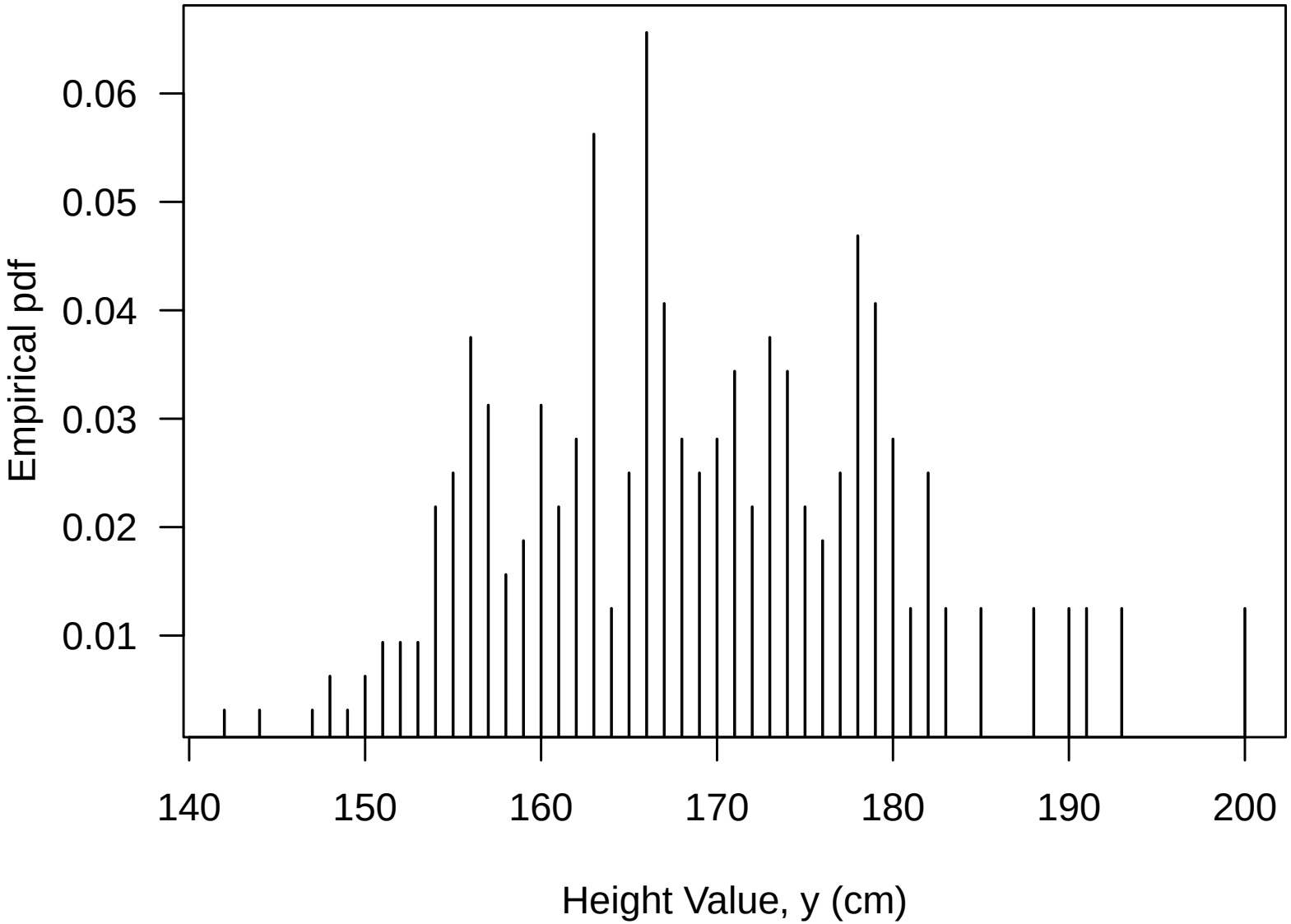
```
# use function emppmf to calculate pmf
strresult <- emppmf(htstrat$height,htstrat$sampwt)
print(as_tibble(head(strresult)), n = 6)
```

```
## # A tibble: 45 x 2
##    vals      epmf
##   <dbl> <dbl[1d]>
## 1   142   0.00312
## 2   144   0.00312
## 3   147   0.00312
## 4   148   0.00625
## 5   149   0.00312
## 6   150   0.00625
## # i 39 more rows
```

```
# plot
```

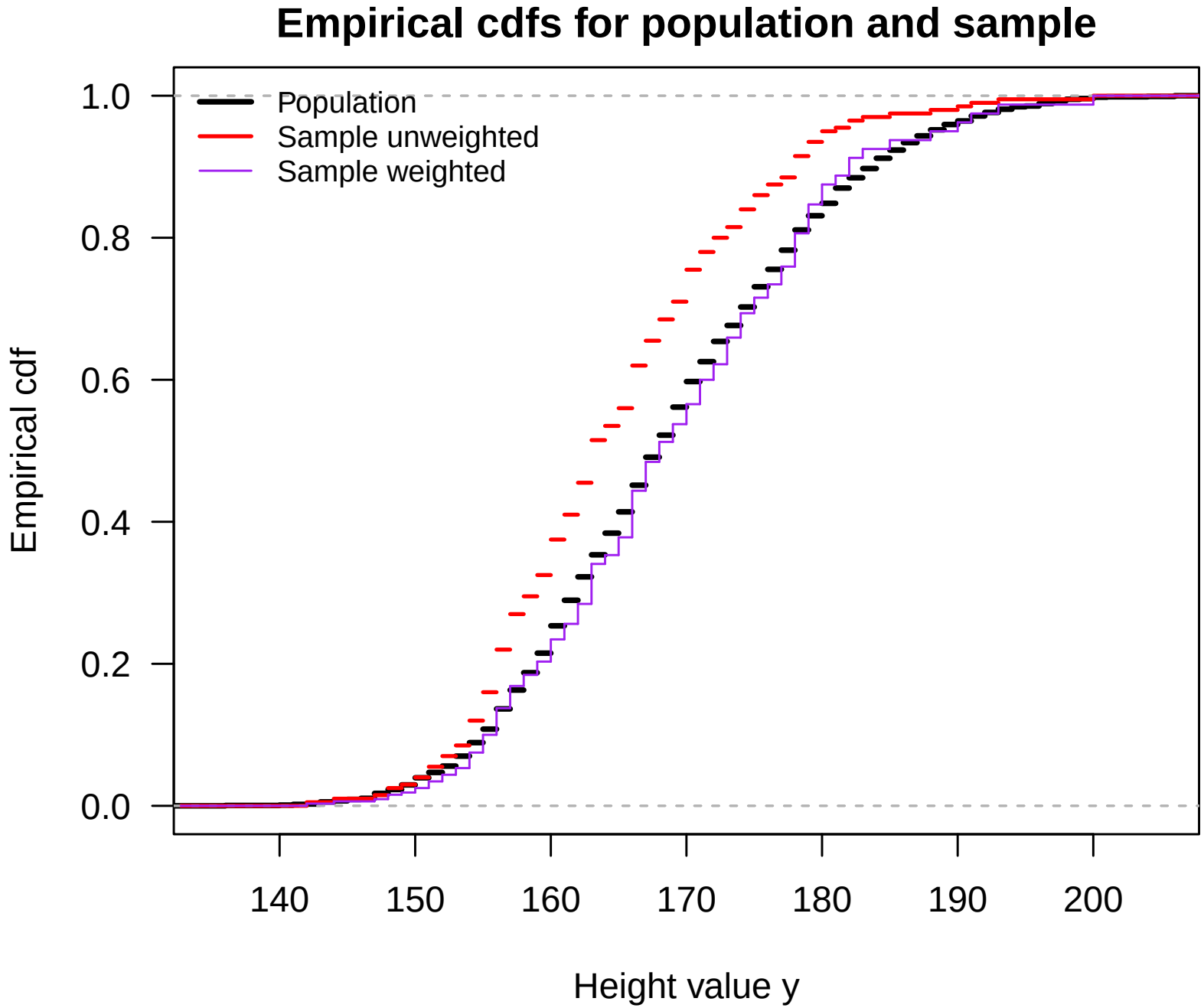
```
par(las=1)
plot(strresult$vals, strresult$epmf,type="h",
 xlab="Height Value, y (cm)",
 ylab="Empirical pdf",lwd=1.2,
 main="Empirical pdf for stratified sample of heights (weighted)")
```

## Empirical pdf for stratified sample of heights (weighted)



```r
# Empirical cdf of height for data htpop, and for data htstrat,
# with and without weights
d0710 <- svydesign(id = ~1, strata = ~gender,
                   fpc = c(rep(1000,160),rep(1000,40)),
                   data = htstrat)
cdf.weighted<-svycdf(~height, d0710)
cdf.weighted

## Weighted ECDFs: svycdf(~height, d0710)

## evaluate the function for height 144
cdf.weighted[[1]](144)

## [1] 0.00625

## compare to population and unweighted sample ecdfs.
cdf.pop<-ecdf(htpop$height)     # ecdf for population
cdf.samp<-ecdf(htstrat$height) # unweighted ecdf of sample
par(las=1,mar=c(5.1,4.1,2.1,2.1))
plot(cdf.pop, do.points = FALSE,
     xlab="Height value y",ylab="Empirical cdf",xlim=c(135,205),
```

```
     lwd=2.5,
     main="Empirical cdfs for population and sample")
lines(cdf.samp, col="red", do.points = FALSE, lwd=1.8)
lines(cdf.weighted[[1]], do.points = FALSE,  col ="purple",lwd=1)
legend("topleft", legend=c("Population", "Sample unweighted",
    "Sample weighted"),col=c("black", "red", "purple"),
    lwd=c(2.5,1.8,1),cex=0.8,bty="n")
```


Empirical cdfs for population and sample
Population
Sample unweighted
Sample weighted
Empirical cdf
1.0
0.8
0.6
0.4
0.2
0.0
140
150
160
170
180
190
200
Height value y

## Examples 7.5, 7.6, and 9.12

Table 1: Estimates from population and different samples

| Quantity | Population | SRS | Stratified no weights | Stratified with weights |
|---|---|---|---|---|
| Mean | 168.6 | 168.9 | 164.6 | 169.0 |
| Median | 168 | 169 | 163 | 168 |
| 25th percentile | 160 | 160 | 157 | 161 |
| 90th percentile | 184 | 184 | 178 | 182 |
| variance | 124.5 | 122.6 | 93.4 | 116.8 |

```
##### Example 7.5

data(htsrs)
dhtsrs<-svydesign(id = ~1,weights=rep(2000/200,200),
    fpc=rep(2000,200), data=htsrs)
# cdf treated as step function, gives values in Table 7.1 of SDA
svyquantile(~height, dhtsrs, quantiles=c(0.25,0.5,0.75,0.9),
            ties = "discrete")

## $height
##      quantile ci.2.5 ci.97.5        se
## 0.25      160    159     163 1.0142211
## 0.5       169    168     171 0.7606658
## 0.75      176    174     179 1.2677764
## 0.9       184    182     187 1.2677764
##
## attr(,"hasci")
## [1] TRUE
## attr(,"class")
## [1] "newsvyquantile"

# interpolated quantiles (usually preferred method)
svyquantile(~height, dhtsrs, quantiles=c(0.25,0.5,0.75,0.9),
```

```
            ties = "rounded")

## $height
##      quantile ci.2.5 ci.97.5        se
## 0.25      160    159     163 1.0142211
## 0.5       169    168     171 0.7606658
## 0.75      176    174     179 1.2677764
## 0.9       184    182     187 1.2677764
##
## attr(,"hasci")
## [1] TRUE
## attr(,"class")
## [1] "newsvyquantile"
```

```
##### Examples 7.6 and 9.12

data(htstrat)
popsize_recode <- c('F' = 1000, 'M' = 1000)
# create a new variable popsize for population size
htstrat$popsize<-popsize_recode[htstrat$gender]
head(as.data.frame(htstrat))
```

```
##    rn height gender popsize
## 1 201    166      F    1000
## 2 965    163      F    1000
## 3 490    166      F    1000
## 4 249    155      F    1000
## 5 260    154      F    1000
## 6 324    160      F    1000
```

```r
# design object
# svydesign calculates the weights here from the fpc argument
dhtstrat<-svydesign(id = ~1, strata = ~gender,
    fpc = ~popsize,data = htstrat)
# ties = "discrete" gives values in Table 7.1 of SDA
svyquantile(~height, dhtstrat, c(0.25,0.5,0.75,0.9),
            ties = "discrete")
```

```
## $height
##      quantile ci.2.5 ci.97.5        se
## 0.25      161    160     163 0.7606423
## 0.5       168    166     171 1.2677372
## 0.75      177    174     179 1.2677372
## 0.9       182    180     191 2.7890219
##
```

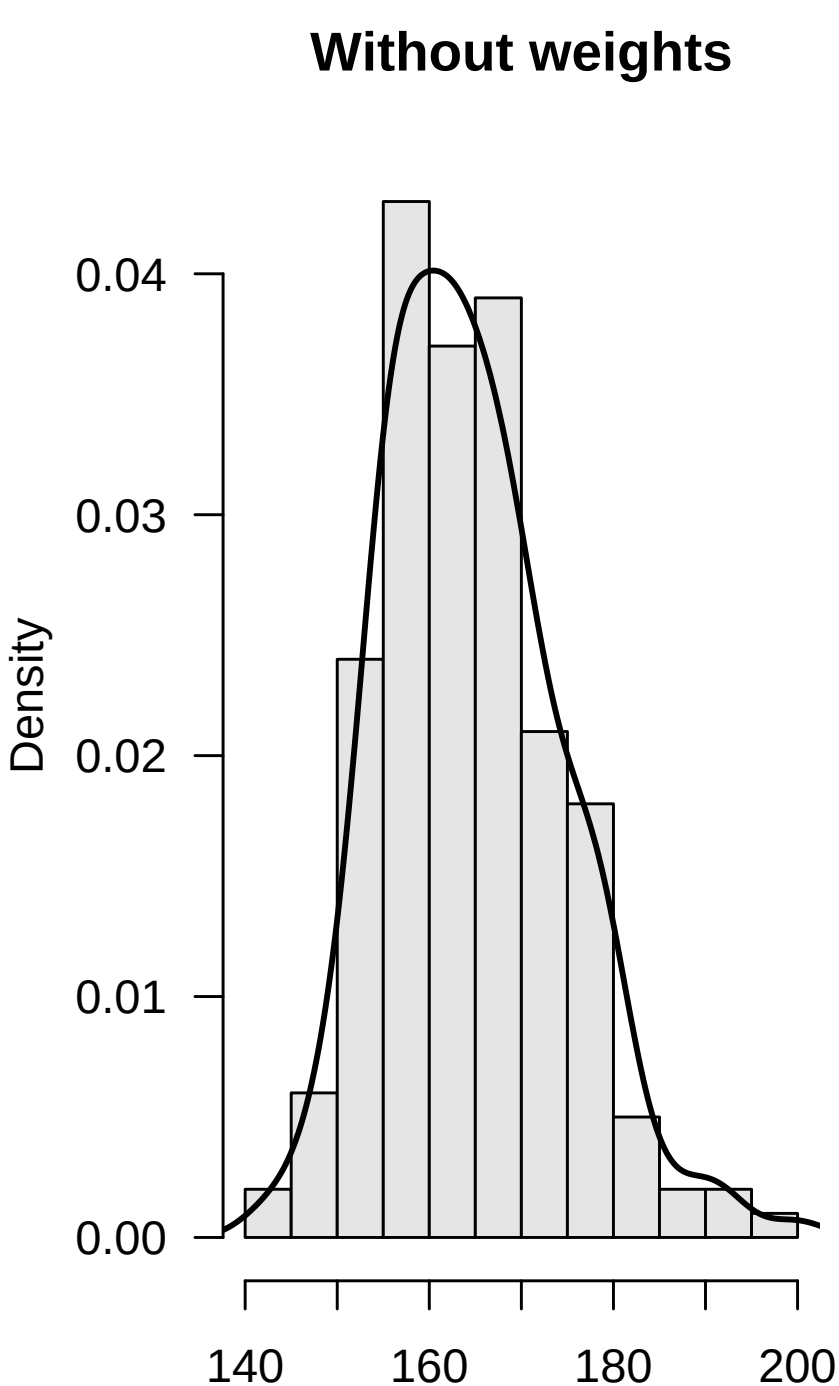
```
## attr(,"hasci")
## [1] TRUE
## attr(,"class")
## [1] "newsvyquantile"

# ties = "rounded" gives values in Example 9.12 of SDA
svyquantile(~height, dhtstrat, c(0.25,0.5,0.75,0.9),
            ties = "rounded",
            ci=TRUE, interval.type = "beta")

## $height
##      quantile ci.2.5 ci.97.5        se
## 0.25      161    160     163 0.7606423
## 0.5       168    166     171 1.2677372
## 0.75      177    174     179 1.2677372
## 0.9       182    179     190 2.7890219
##
## attr(,"hasci")
## [1] TRUE
## attr(,"class")
## [1] "newsvyquantile"
```

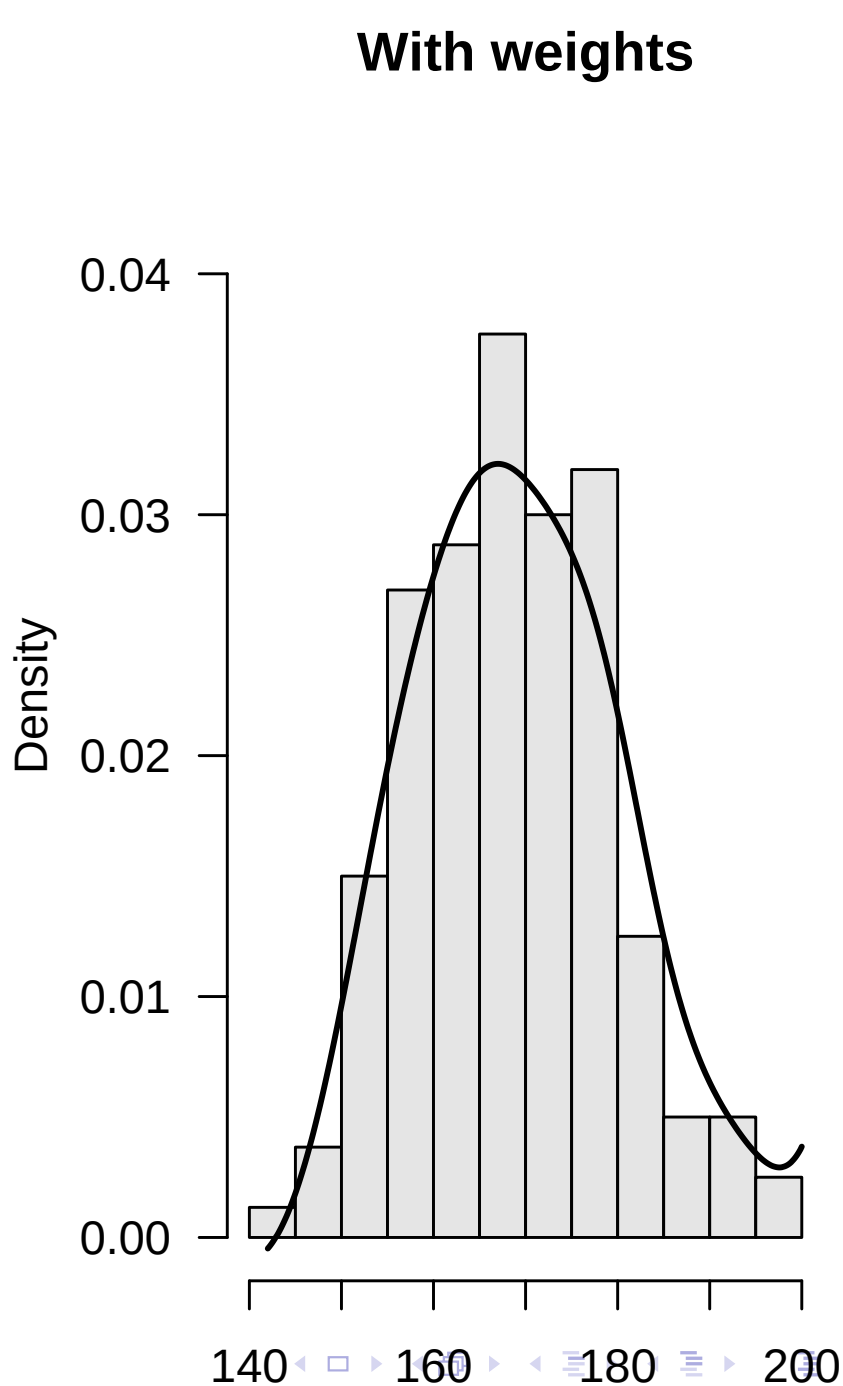
**Examples 7.6 and 9.12**

- data: htstrat
- Histogram with smoothed density function
- Boxplots

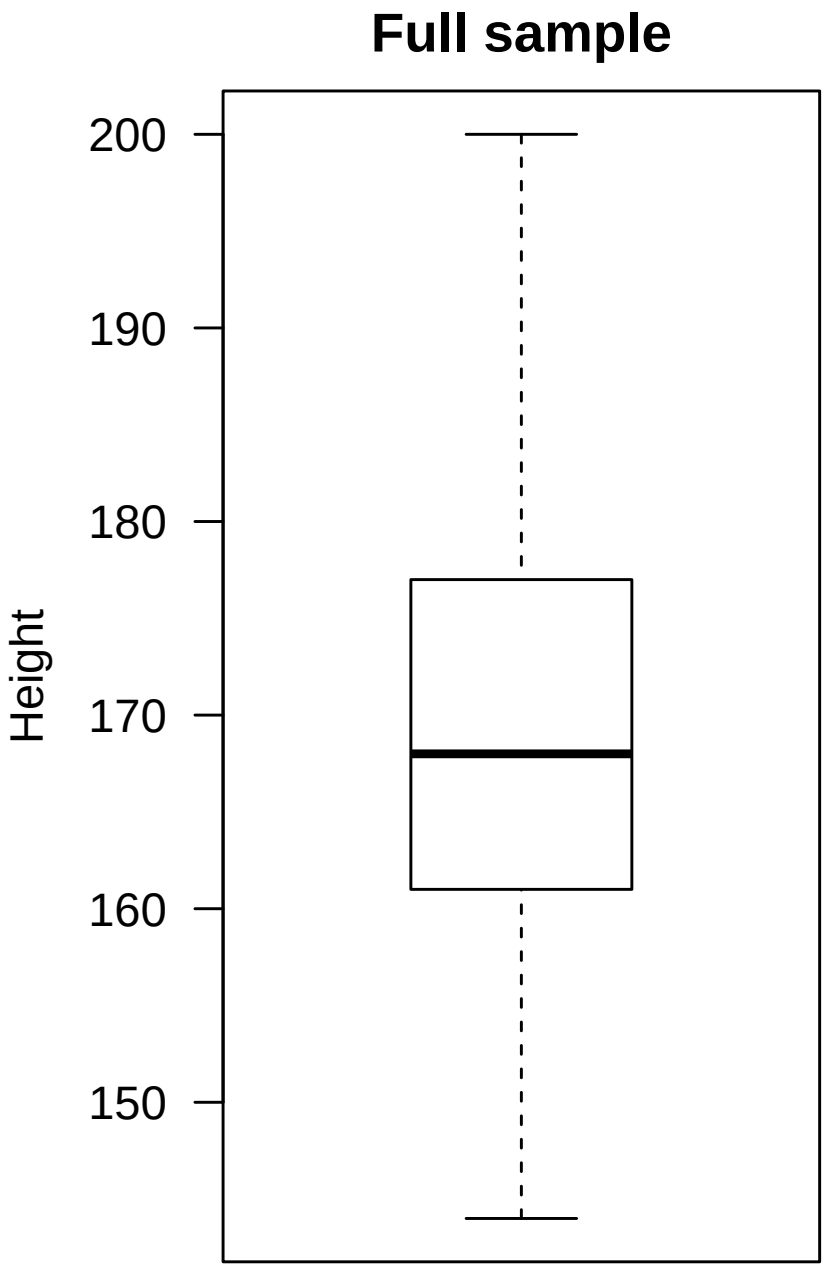
```r
##### Examples 7.6 and 9.12
# histogram and smoothed density function
data(htstrat)
# set graphics parameters, 1*2 plots, axis labels horizontal
par(mfrow=c(1,2),las=1,mar=c(2.1,4.1,2.1,0.3))
# Histogram overlaid with kernel density curve
# (without weight information)
# Displays the sample values, but does not estimate
# population histogram
# freq=FALSE changes the vertical axis to density
# breaks tell how many breakpoints to use
hist(htstrat$height,main="Without weights", xlab = "Height (cm)",
     breaks = 10, col="gray90", freq=FALSE,
     xlim=c(140,200), ylim=c(0,0.045))
# overlaid with kernel density curve
lines(density(htstrat$height),lty=1,lwd=2)
# Histogram (with weight information)
# create survey design object, weights calculated from fpc here
d0710 <- svydesign(id = ~1, strata = ~gender,
                   fpc = c(rep(1000,160),rep(1000,40)),
                data = htstrat)
d0710
```

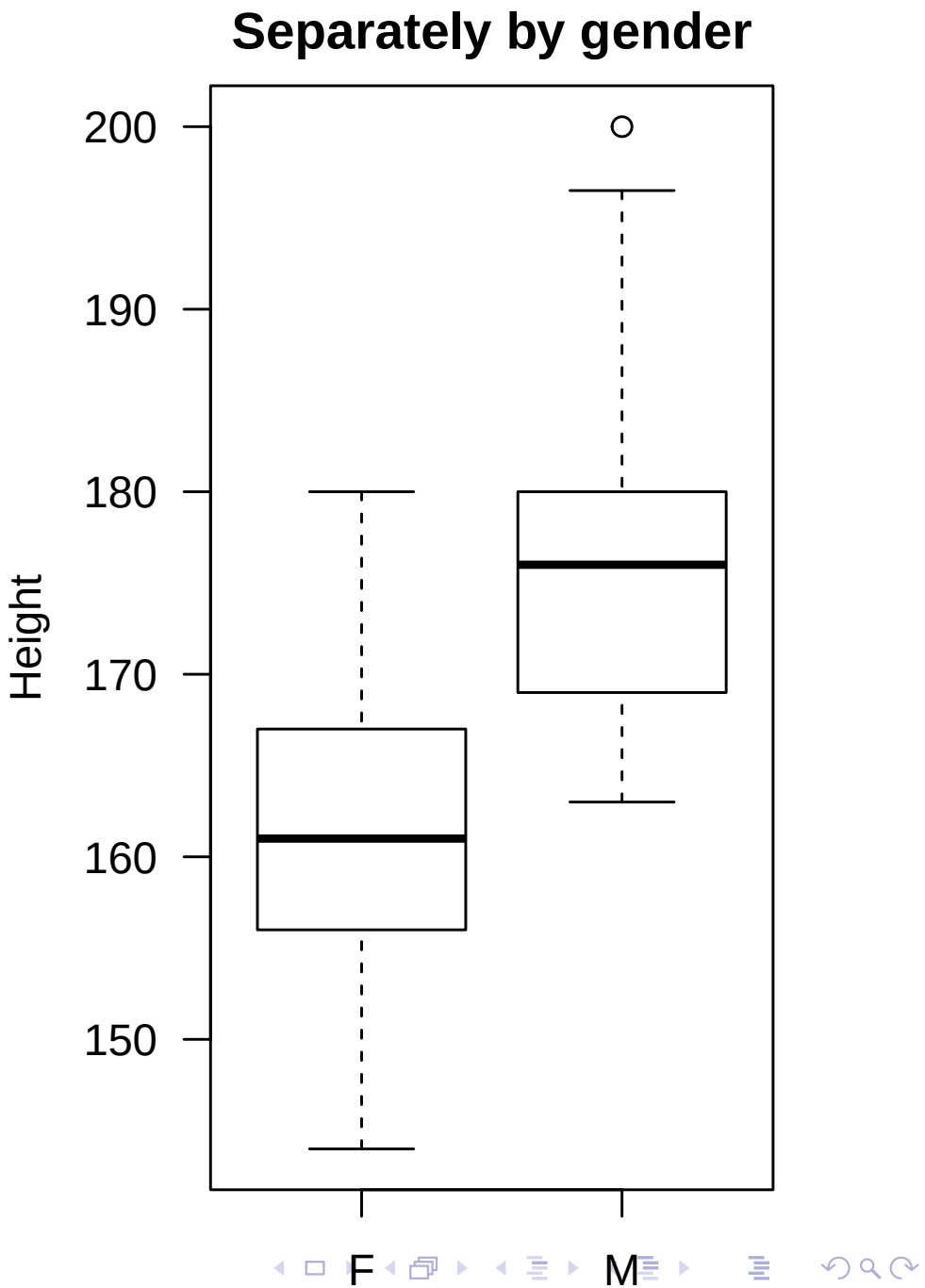
```
## Stratified Independent Sampling design
## svydesign(id = ~1, strata = ~gender, fpc = c(rep(1000, 160),
##     rep(1000, 40)), data = htstrat)

svyhist(~height,d0710, main="With weights",xlab = "Height (cm)",
        breaks = 10, col="gray90", freq=FALSE,
        xlim=c(140,200), ylim=c(0,0.045))
dens1<-svysmooth(~height,d0710,bandwidth=5)
lines(dens1,lwd=2) # draw the density line
```


Without weights
Density
0.04
0.03
0.02
0.01
0.00
140
160
180
200


With weights
Density
0.04
0.03
0.02
0.01
0.00
140
160
180
200

```
# boxplot
par(mfrow=c(1,2),las=1,mar=c(2.1,4.1,2.1,0.3))
# boxplot (with weight information)
svyboxplot(height~1,d0710,ylab="Height",xlab=" ",
           main="Full sample")
svyboxplot(height~gender,d0710,ylab="Height",xlab="Gender",
           main="Separately by gender")
```


Full sample
Height
200
190
180
170
160
150


Separately by gender
Height
200
190
180
170
160
150
F
M

**Example 7.9:** National Health and Nutrition Examination Survey (NHANES), (Centers for Disease Control and Prevention, 2017). In this example, we look at statistics about body mass index (BMI, variable bmxbmi ) for adults age 20 and over and plots.

- The *weights* argument in *svydesign* specifies the variable containing the final weights.
  *wtint2yr*: weight for the set of persons with interview data, *wtmec2yr*: weight for the subset of interviewed persons who had a medical examination.
  BMI is measured in the medical examination, so the appropriate weight variable to use is *wtmec2yr*.
- The *strata* and *id* arguments are used as before, except now we include both of them. The `strata=~sdmvstra` argument says that *sdmvstra* is the variable giving the stratum membership. The psus specified in `id=~sdmvpsu` are the first-stage sampling units.

- In the NHANES data, the two psus in each stratum are labeled as '1' and '2'. The `nest=TRUE` argument says that psu labels are nested within strata—that is, multiple strata have the same psu labels. Typing `nest=TRUE` ensures that psu 1 in stratum 1 is recognized as being a different psu than psu 1 in stratum 2.
  When there is one stratum, the results are the same if you have `nest=TRUE`, `nest=FALSE`, or simply omit the *nest* argument.
- No *fpc* argument is included in *svydesign*. With complex samples such as NHANES, we usually want to calculate the with-replacement variance, which requires only psu-level information.

```r
### Computing Estimates from Stratified Multistage Samples ####
# note: load package in this order to get correct data nhanes
# library(survey)
# library(SDAResources)
data(nhanes)
nrow(nhanes) #9971
```

```
## [1] 9971
```

```r
names(nhanes)
```

```
##  [1] "sdmvstra" "sdmvpsu"  "wtint2yr" "wtmec2yr" "ridstatr" "ridageyr"
##  [7] "ridagemn" "riagendr" "ridreth3" "dmdeduc2" "dmdfmsiz" "indfmpir"
## [13] "bmxwt"    "bmxht"    "bmxbmi"   "bmxwaist" "bmxleg"   "bmxarml"
## [19] "bmxarmc"  "bmdavsad" "lbxtc"    "bpxpls"   "sbp"      "dbp"
## [25] "bpread"
```

```r
# count number of observations with missing value
# for ridageyr, bmxbmi
sum(is.na(nhanes$ridageyr)) # ridageyr gives age in years
```

```
## [1] 0
```

```r
sum(is.na(nhanes$bmxbmi))   # bmxbmi gives BMI
```

```
## [1] 1215
```

- *subset* function: define domains. This specifies that estimates are desired for the domain of persons age 20 and older having data for BMI (with *age20d*=1), and carries the stratification and clustering information from the full design over for analyzing the subset.
- If you just created a subset of the data consisting of the observations having *ridageyr* $\geq 20$, in some instances (for example, when some psus have no members of the domain), the standard errors would be incorrect; by using the *subset* function, the correct standard errors are calculated.
- We exclude adults with missing values of *bmxbmi* from the domain of interest with *age20d*=1. The estimates are computed from the adults who have data. If the domain contained missing values, we would need to include option `na.rm=TRUE` in the *svymean* function to be able to calculate statistics.

```r
# define age20d
nhanes$age20d<-rep(0,nrow(nhanes))
nhanes$age20d[nhanes$ridageyr >=20 & !is.na(nhanes$bmxbmi)]<-1
# check missing value counts for new variables
sum(is.na(nhanes$age20d))
```

```
## [1] 0
```

```r
sum(nhanes$age20d) # how many records in domain?
```

```
## [1] 5406
```

```r
head(nhanes[, c(1:4, 15, 26)])
```

```
## # A tibble: 6 x 6
##   sdmvstra sdmvpsu wtint2yr wtmec2yr bmxbmi age20d
##      <dbl>   <dbl>    <dbl>    <dbl>  <dbl>  <dbl>
## 1      125       1  134671.  135630.   27.8      1
## 2      125       1   24329.   25282.   30.8      1
## 3      131       1   12400.   12576.   28.8      1
## 4      131       1  102718.  102079.   42.4      1
## 5      126       2   17628.   18235.   20.3      1
## 6      128       1   11252.   10879.   28.6      1
```

```r
# stratified cluster design
d0709 <- svydesign(id = ~sdmvpsu, strata=~sdmvstra,
  weights=~wtmec2yr, nest=TRUE, data = nhanes)
# domain estimation, age20+
d0709sub<-subset(d0709, age20d ==1)
d0709sub
```

```
## Stratified 1 - level Cluster Sampling design (with replacement)
## With (30) clusters.
## subset(d0709, age20d == 1)
```

```r
# Request means and design effects
nhmeans<-svymean(~bmxbmi, d0709sub, deff=TRUE)
degf(d0709sub)
```

```
## [1] 15
```

```r
nhmeans
```

```
##            mean     SE   DEff
## bmxbmi 29.3891 0.2532 7.1248
```

```r
confint(nhmeans,df=degf(d0709sub))
```

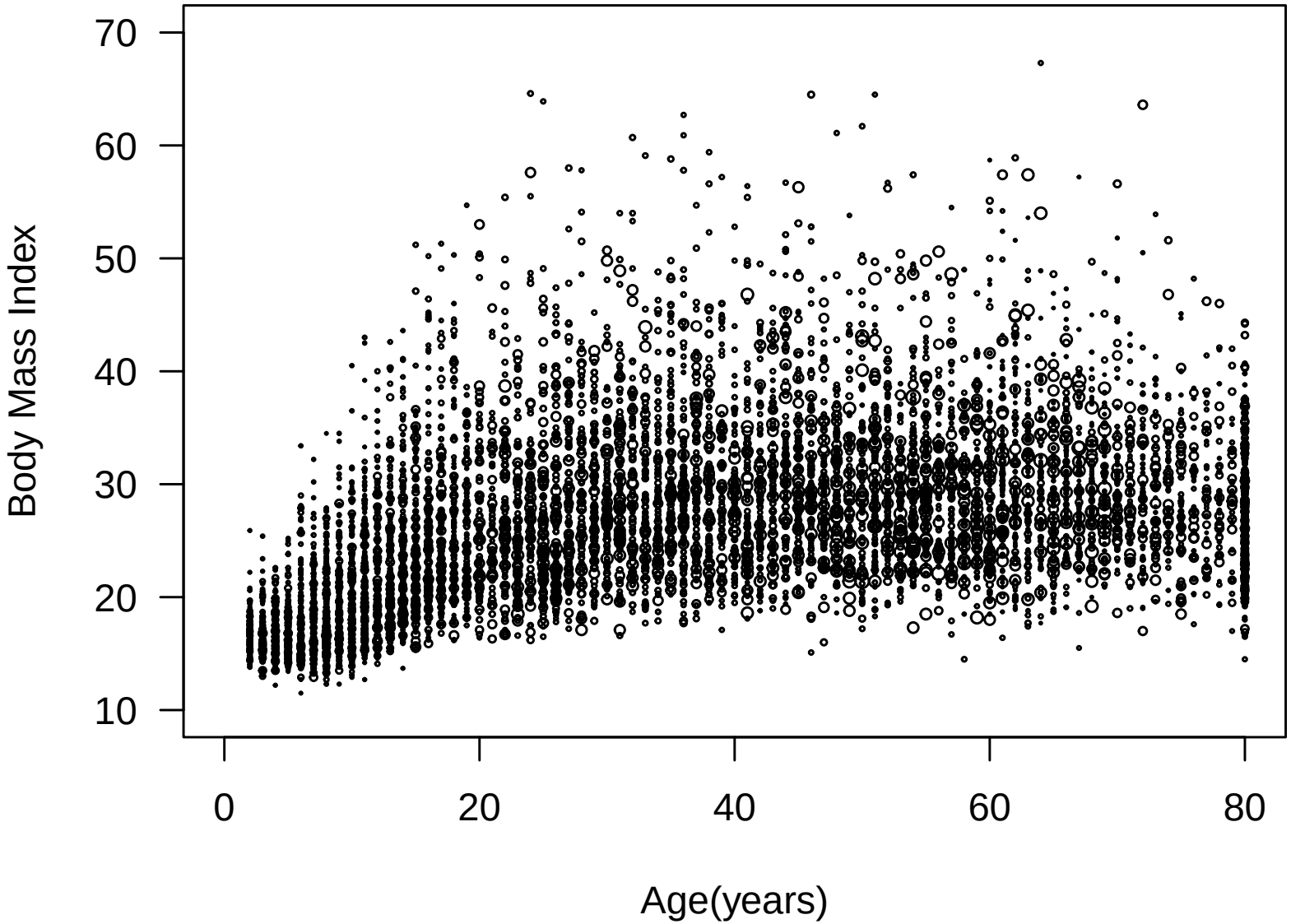
```
##            2.5 %   97.5 %
## bmxbmi 28.84942 29.92878
```

The mean BMI for adults aged 20 and over is **29.389**, calculated using the design object *d0709sub*, with a 95% confidence interval of [28.849, 29.929].

Next, we present bubble plots along with a smoothed trend line. The area of each circle in the plot is proportional to the sum of the weights for the sample observations.

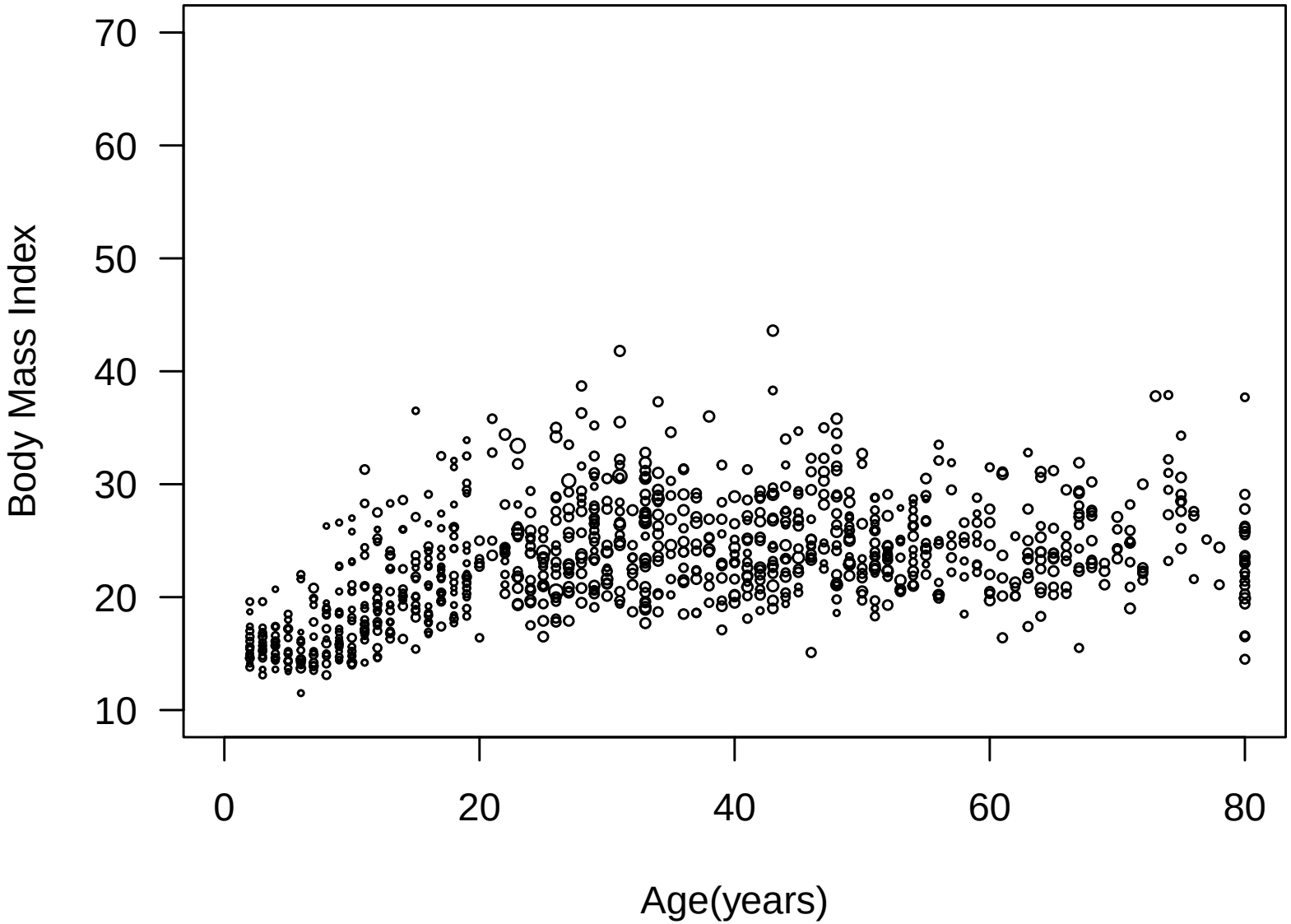
```
### Bubble plots
par(las=1) # make tick mark labels horizontal
svyplot(bmxbmi~ridageyr, design=d0709, style="bubble", inches=0.03,
        xlab="Age(years)",ylab="Body Mass Index",
        xlim=c(0,80),ylim=c(10,70),
        main="Weighted bubble plot of BMI versus age")
```

## Weighted bubble plot of BMI versus age



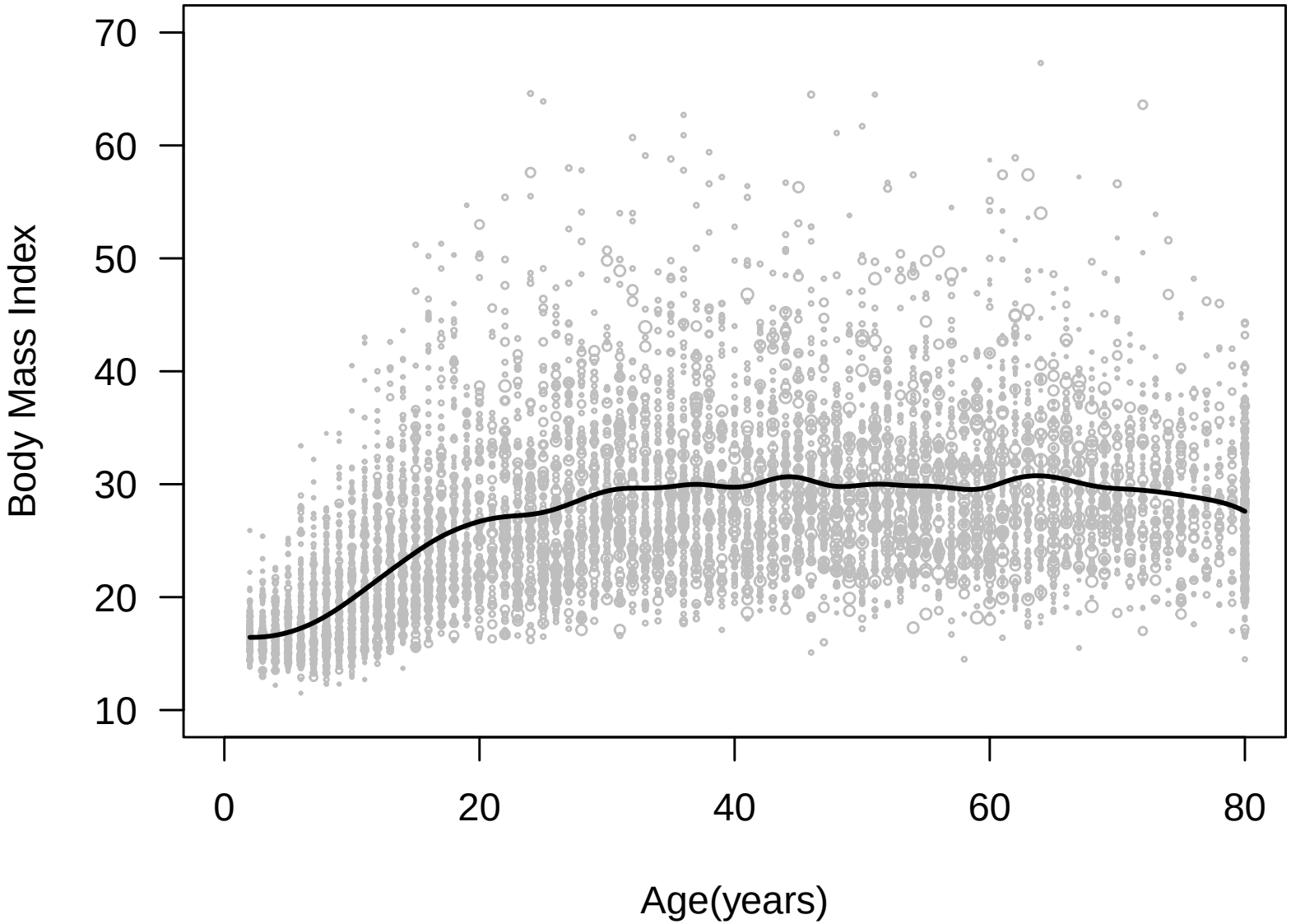
```
### Plot data for a domain
# define subset
d0709subA<-subset(d0709, ridreth3==6)
par(las=1) # make tick mark labels horizontal
svyplot(bmxbmi~ridageyr, design=d0709subA,
        style="bubble",inches = 0.03,
        xlab="Age(years)",ylab="Body Mass Index",
        xlim=c(0,80),ylim=c(10,70),
main="Weighted bubble plot of BMI versus age for Asian Americans")
```

## Weighted bubble plot of BMI versus age for Asian Americans



```r
### Smoothed trend line for mean
# Smoothed trend line with bubble plot of BMI versus age
# plot data bmxbmi~ridageyr
par(las=1) # make tick mark labels horizontal
svyplot(bmxbmi~ridageyr, design=d0709, style="bubble",
        basecol="gray",inches=0.03,
        xlab="Age(years)",ylab="Body Mass Index",
        xlim=c(0,80),ylim=c(10,70),
main="Smoothed trend line with bubble plot of BMI versus age")
# plot smoothing trend line
# library(KernSmooth)  # install and load the package
smth<-svysmooth(bmxbmi~ridageyr,d0709)
lines(smth,lwd=2)
```

## Smoothed trend line with bubble plot of BMI versus age



# Disclaimer

Slides are intended for a course based on the book: *Sampling: Design and Analysis, Third Edition* by Sharon L. Lohr
All examples, datasets, and pages directly scanned and included in this chapter are from the textbook.
References to papers or books cited in these slides can be found in the Bibliography section of the textbook.