260677676_Assignment_4 -_MATH_525

Dan Yunheum Seol

2019 4 20

6.6

The file azcounties.dat gives data from the 2000 U.S. Census on population and housing unit counts for the counties in Arizona (excluding Maricopa County and Pima County, which are much larger than the other counties and would be placed in a separate stratum $t = \sum_{i=1}^{1} 3t_i$. The file has the value of t_i for every county so you can calculate the population total and variance.

a.

Calculate the selection probabilities ψ_i for a sample of size 1 with probability proportional to 2000 population. Find $\widehat{t}_\psi$ for each possible sample, and calculate the theoretical variance $V(\widehat{t}_\psi)$

b.

Repeat (a) for an equal probability sample of size 1. How do the variances com- pare? Why do you think one design is more efficient than the other.

We have the following formulae that we can refer to:

$$\widehat{t}_\psi = \frac{1}{n} \sum_{i \in R} \frac{t_i}{\psi_i}$$

$$V(\widehat{t}_\psi) = \frac{1}{n} \sum_{i=1}^{N} \psi_i [\frac{t_i}{\psi_i} - t]^2$$

and

$$\widehat{V}(\widehat{t}_\psi) = \frac{1}{n(n-1)} \sum_{i=1}^{N} \psi_i [\frac{t_i}{\psi_i} - \widehat{t}_\psi]^2$$

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------

## v ggplot2 3.1.0     v purrr   0.3.0
## v tibble  2.0.1     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.3.1     v forcats 0.3.0

## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(survey)
```

```
## Loading required package: grid

## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following object is masked from 'package:tidyr':
##
##     expand

## Loading required package: survival

##
## Attaching package: 'survey'

## The following object is masked from 'package:graphics':
##
##     dotchart
```

```r
library(srvyr)
```

```
##
## Attaching package: 'srvyr'

## The following object is masked from 'package:stats':
##
##     filter
```

```r
library(pps)
library(sampling)
```

```
##
## Attaching package: 'sampling'

## The following objects are masked from 'package:survival':
##
##     cluster, strata
```

```r
azcounties <- read_csv('azcounties.csv')
```

```
## Parsed with column specification:
## cols(
##   number = col_double(),
##   name = col_character(),
##   population = col_double(),
##   housing = col_double()
## )
```

```r
#azcounties_clean <-azcounties %>% drop_na(population)
dim(azcounties)
```

```
## [1] 13  4
```

```r
#dim(azcounties_clean)
#No missing data
head(azcounties)
```

```
## # A tibble: 6 x 4
##   number name     population housing
```

```
##    <dbl> <chr>         <dbl>   <dbl>
## 1      1 Apache        69423   31621
## 2      2 Cochise      117755   51126
## 3      3 Coconino     116320   53443
## 4      4 Gila          51335   28189
## 5      5 Graham        33489   11430
## 6      6 Greenlee       8547    3744
```

```
Mi_az <- azcounties %>% group_by(name) %>% summarise(Mi = population,ti=housing)  %>% ungroup() %>% muta

Mi_az %>% head()
```

```
## # A tibble: 6 x 4
##   name         Mi    ti phi_eq
##   <chr>     <dbl> <dbl>  <dbl>
## 1 Apache    69423 31621 0.0769
## 2 Cochise  117755 51126 0.0769
## 3 Coconino 116320 53443 0.0769
## 4 Gila      51335 28189 0.0769
## 5 Graham    33489 11430 0.0769
## 6 Greenlee   8547  3744 0.0769
```

```
Mi_az = Mi_az %>% mutate(psi_k =Mi/sum(Mi))
Mi_az
```

```
## # A tibble: 13 x 5
##    name           Mi    ti phi_eq   psi_k
##    <chr>       <dbl> <dbl>  <dbl>   <dbl>
##  1 Apache      69423 31621 0.0769 0.0572
##  2 Cochise    117755 51126 0.0769 0.0969
##  3 Coconino   116320 53443 0.0769 0.0958
##  4 Gila        51335 28189 0.0769 0.0423
##  5 Graham      33489 11430 0.0769 0.0276
##  6 Greenlee     8547  3744 0.0769 0.00704
##  7 La Paz      19715 15133 0.0769 0.0162
##  8 Mohave     155032 80062 0.0769 0.128
##  9 Navajo      97470 47413 0.0769 0.0802
## 10 Pinal      179727 81154 0.0769 0.148
## 11 Santa Cruz  38381 13036 0.0769 0.0316
## 12 Yavapai    167517 81730 0.0769 0.138
## 13 Yuma       160026 74140 0.0769 0.132
```

```
#a
#is.data.frame(Mi_az)
tHat_psi <- (Mi_az$ti)/(Mi_az$psi_k) #n is 1 so we divide it by 1
t <- sum(Mi_az$ti)
I <- rep(1, 13)

tvec <-  t*I
part1<- Mi_az$psi_k*(tHat_psi-tvec)
part2<- (tHat_psi-tvec)
v.tHat <- part1 %*% part2

print("Actual total")
```

```
## [1] "Actual total"
```

```r
t
```

```
## [1] 572221
```

```r
print("Estimated totals")
```

```
## [1] "Estimated totals"
```

```r
tHat_psi
```

```
##  [1] 553292.1 527405.6 558108.6 667034.6 414597.1 532113.6 932417.7
##  [8] 627317.4 590892.8 548502.8 412582.0 592659.0 562787.3
```

```r
print("Theoretical variance")
```

```
## [1] "Theoretical variance"
```

```r
v.tHat
```

```
##            [,1]
## [1,] 4789282131
```

```r
#b
tHat_phi <- (Mi_az$ti)/(Mi_az$phi_eq) #n is one so we divide it by 1

print("Actual total")
```

```
## [1] "Actual total"
```

```r
t
```

```
## [1] 572221
```

```r
print("Estimated totals - equal probability")
```

```
## [1] "Estimated totals - equal probability"
```

```r
tHat_phi
```

```
##  [1]  411073  664638  694759  366457  148590   48672  196729 1040806
##  [9]  616369 1055002  169468 1062490  963820
```

```r
part3 <-  Mi_az$phi_eq*(tHat_phi-tvec) # Mi_az$phi_eq= 1/13
part4 <- (tHat_phi-tvec)

v.tHat2 <- part3 %*% part4

print("Theoretical variance - equal propbability")
```

```
## [1] "Theoretical variance - equal propbability"
```

```r
v.tHat2
```

```
##              [,1]
## [1,] 130534375140
```

```r
print("unequal prob var/equal prob var")
```

```
## [1] "unequal prob var/equal prob var"
```

```r
v.tHat/v.tHat2
```

```
##            [,1]
## [1,] 0.03668982
```

It seems that the sample with unequal probabilities show a far better performance for both estimated mean and variance. In comparison, the sample from part(a) showed a more consistent fit for total estimates and this is reflected in the reduced variance compared to that for the smaple from part(b) (the variance reduced to one third to that of sample with equal probabilities). This happens since samples with equal probabilities tend to overrepresent counties with small number of population such as Graham or Greenlee (a county with small population has no reason to have excessive number of housing!).

c.

Now take a with-replacement sample of size 3. Find $\widehat{t}_\psi$ and $V(\widehat{t}_\psi)$.

```
set.seed(19970329)
n_az = 3  #sample we have been asked to draw

oneaz_WR <- Mi_az %>% sample_n(size=n_az, replace=T, weight=Mi)
oneaz_WR = oneaz_WR %>% group_by(name) %>% mutate(replication= 1:n())
oneaz_WR
```

```
## # A tibble: 3 x 6
## # Groups:   name [3]
##   name          Mi    ti phi_eq  psi_k replication
##   <chr>      <dbl> <dbl>  <dbl>  <dbl>       <int>
## 1 Pinal     179727 81154 0.0769 0.148            1
## 2 Yuma      160026 74140 0.0769 0.132            1
## 3 Coconino  116320 53443 0.0769 0.0958           1
```

```
oneaz_WR  %>% group_by(name) %>% summarise(count = n()) %>% arrange(desc(count))
```

```
## # A tibble: 3 x 2
##   name      count
##   <chr>     <int>
## 1 Coconino      1
## 2 Pinal         1
## 3 Yuma          1
```

```
oneaz_sample_WR  <-inner_join(azcounties,oneaz_WR, by='name') %>% mutate(weight= 1/(n_az*psi_k))

dim(oneaz_sample_WR)
```

```
## [1]  3 10
```

```
#a
#sample of size 10 with replacement
oneaz_cluster_WR_design <-svydesign(id=~name, data= oneaz_sample_WR,weight=~weight)
print("Estimated total number of housing units")
```

```
## [1] "Estimated total number of housing units"
```

```
svytotal(~housing, oneaz_cluster_WR_design)
```

```
##          total     SE
## housing 556466 4204.5
```

```
print("Actual total number of housing units")
```

```
## [1] "Actual total number of housing units"
```

```
t
```

```
## [1] 572221
```

```
print("Sample variance of estimated total")
```

```
## [1] "Sample variance of estimated total"
```

```
4204.5^2
```

```
## [1] 17677820
```

6.10

Use your sample of states drawn with probability proportional to population, from Exercise 9, for this problem. ## a. Using the sample, estimate the total number of counties in the United States, and find the standard error of your estimate. How does your estimate compare with the true value of total number of counties (which you can calculate, since the file statepps.dat contains the data for the whole population)? ## b. Now suppose that your friend Tom finds the ten values of numbers of counties in your sample, but does not know that you selected these states with probabilities proportional to population. Tom then estimates the total number of counties using formulas for an SRS. What values for the estimated total and its standard error are calculated by Tom? How do these values differ from yours? Is Tom's estimator unbiased for the population total

```
library(tidyverse)
library(survey)
library(srvyr)
library(pps)
library(sampling)
```

```
statepps <- read_csv('statepps.csv')
```

```
## Parsed with column specification:
## cols(
##   state = col_character(),
##   counties = col_double(),
##   cumcount = col_double(),
##   landarea = col_double(),
##   cumland = col_double(),
##   popn = col_double(),
##   cumpopn = col_double()
## )
```

```
#statepps_clean <-statepps %>% drop_na(popn,counties)
#dim(statepps)
#dim(statepps_clean)
# both have dimension 51 7 -no missing data

head(statepps)
```

```
## # A tibble: 6 x 7
##   state      counties cumcount landarea cumland    popn  cumpopn
##   <chr>         <dbl>    <dbl>    <dbl>   <dbl>   <dbl>    <dbl>
## 1 Alabama          67       67    50750   50750 4137511  4137511
## 2 Alaska           25       92   570374  621124  587766  4725277
## 3 Arizona          15      107   113642  734766 3832368  8557645
```

```
## 4 Arkansas         75      182    52075   786841   2394253 10951898
## 5 California       58      240   155973   942814 30895356 41847254
## 6 Colorado         63      303   103729 1046543   3464675 45311929
```

```r
#statepps
```

```r
Mi_tbl <- statepps %>% group_by(state) %>% summarise(Mi = popn) %>% ungroup() %>% mutate(N = n())

Mi_tbl %>% head()
```

```
## # A tibble: 6 x 3
##   state            Mi     N
##   <chr>         <dbl> <int>
## 1 Alabama     4137511    51
## 2 Alaska       587766    51
## 3 Arizona     3832368    51
## 4 Arkansas    2394253    51
## 5 California 30895356    51
## 6 Colorado    3464675    51
```

```r
Mi_tbl = Mi_tbl %>% mutate(psi_k =Mi/sum(Mi))
Mi_tbl
```

```
## # A tibble: 51 x 4
##    state                      Mi     N   psi_k
##    <chr>                   <dbl> <int>   <dbl>
##  1 Alabama               4137511    51 0.0162
##  2 Alaska                 587766    51 0.00230
##  3 Arizona               3832368    51 0.0150
##  4 Arkansas              2394253    51 0.00939
##  5 California           30895356    51 0.121
##  6 Colorado              3464675    51 0.0136
##  7 Connecticut           3279116    51 0.0129
##  8 Delaware               690884    51 0.00271
##  9 District of Columbia   585221    51 0.00229
## 10 Florida              13482716    51 0.0529
## # ... with 41 more rows
```

```r
set.seed(19970329)
n = 10  #sample we have been asked to draw

onestage_WR <- Mi_tbl %>% sample_n(size=n, replace=T, weight=Mi)
onestage_WR = onestage_WR %>% group_by(state) %>% mutate(replication= 1:n())
onestage_WR
```

```
## # A tibble: 10 x 5
## # Groups:   state [10]
##    state                Mi     N  psi_k replication
##    <chr>             <dbl> <int>  <dbl>       <int>
##  1 California     30895356    51 0.121            1
##  2 Pennsylvania   11995405    51 0.0470           1
##  3 Missouri        5190719    51 0.0203           1
##  4 Florida        13482716    51 0.0529           1
##  5 North Carolina  6836333    51 0.0268           1
##  6 Oklahoma        3205234    51 0.0126           1
##  7 New York       18109491    51 0.0710           1
```

```
##   8 Maryland        4917269    51 0.0193           1
##   9 Georgia         6773364    51 0.0266           1
##  10 Virginia        6394481    51 0.0251           1
```

```
onestage_WR  %>% group_by(state) %>% summarise(count = n()) %>% arrange(desc(count))
```

```
## # A tibble: 10 x 2
##    state          count
##    <chr>          <int>
##  1 California         1
##  2 Florida            1
##  3 Georgia            1
##  4 Maryland           1
##  5 Missouri           1
##  6 New York           1
##  7 North Carolina     1
##  8 Oklahoma           1
##  9 Pennsylvania       1
## 10 Virginia           1
```

```
onestage_sample_WR  <-inner_join(statepps,onestage_WR, by='state') %>% mutate(weight= 1/(n*psi_k))

dim(onestage_sample_WR)
```

```
## [1] 10 12
```

```
#a
#sample of size 10 with replacement
onestage_cluster_WR_design <-svydesign(id=~state, data= onestage_sample_WR,weight=~weight)
print("Estimated total number of counties")
```

```
## [1] "Estimated total number of counties"
```

```
svytotal(~counties, onestage_cluster_WR_design)
```

```
##            total     SE
## counties 3221.2 753.56
```

```
print("Actual total number of countries")
```

```
## [1] "Actual total number of countries"
```

```
sum(statepps$counties)
```

```
## [1] 3142
```

```
#b
#SRSWR to compare
set.seed(19970319)
onestage_WR_srs <- sample_n(Mi_tbl, size=10, replace=TRUE, weight=rep(1, nrow(Mi_tbl)))
onestage_wr_sample_srs <- inner_join(statepps, onestage_WR_srs, by="state")
dim(onestage_wr_sample_srs)
```

```
## [1] 10 10
```

```
onestage_cluster_WR_srs_design <- svydesign(id =~state, data=onestage_wr_sample_srs, weight=~I(N/n))
svytotal(~counties, onestage_cluster_WR_srs_design)
```

```
##           total     SE
## counties  2601 609.56
```

#6.22

a.

$$E[\widehat{t_y}] = E[\sum_{i=1}^{N} Z_i \frac{u_i}{\pi_i}] = \sum_{i=1}^{N} E[Z_i \frac{u_i}{\pi_i}] = \sum_{i=1}^{N} E[Z_i] \frac{u_i}{\pi_i} = \sum_{i=1}^{N} \pi_i \frac{u_i}{\pi_i} = \sum_{i=1}^{N} u_i =$$

$$\sum_{i=1}^{N}\sum_{k=1}^{M} \frac{l_{ik} y_k}{L_k} = \sum_{k=1}^{M}\sum_{i=1}^{N} \frac{l_{ik} y_k}{L_k} = \sum_{k=1}^{M}\sum_{i=1}^{N} \frac{l_{ik} y_k}{L_k} = \sum_{k=1}^{M} \frac{y_k}{L_k} \sum_{i=1}^{N} l_{ik} = \sum_{k=1}^{M} \frac{y_k}{L_k} L_k = \sum_{k=1}^{M} y_k = t_y$$

So we have an unbiased estimator. I.e. $E[\widehat{t_y}] = t_y$

Also,

$$Var(\widehat{t_y}) = Var(\sum_{i=1}^{N} Z_i \frac{u_i}{\pi_i}) = \sum_{i=1}^{N} \frac{Var(Z_i)}{\pi_i^2} u_i^2 + \sum_{i=1}^{N}\sum_{k \neq i}^{M} \frac{u_i u_k}{\pi_i \pi_k} Cov(Z_i, Z_k)$$

Since Z_i is a sample indicator random variable,

$$Var(Z_i) = \pi_i(1 - \pi_i)$$

.

Similarly,

$$Cov(Z_i, Z_j) = E[Z_i Z_j] - E[Z_i]E[Z_j] = \pi_{ij} - \pi_i \pi_j$$

$$Var(\widehat{t_y}) = \sum_{i=1}^{N} \frac{\pi_i(1 - \pi_i)}{\pi_i^2} u_i^2 + \sum_{i=1}^{N}\sum_{k \neq i}^{M} \frac{(\pi_{ij} - \pi_i \pi_j)}{\pi_i \pi_k} u_i u_k$$

b.

$$\widehat{t_y} = \sum_{k \in S^B} \frac{1}{L^k} \sum_{i=1}^{N} \frac{Z_i}{\pi_i} l_{ik} y_k$$

with weight

$$w_k = \frac{1}{L^k} \sum_{i=1}^{N} \frac{Z_i}{\pi_i} l_{ik} = \sum_{i=1}^{N} \frac{Z_i}{\pi_i} l_{ik} \frac{1}{L_k}$$

while we know $\forall k \notin S^B$

$$\sum_{i=1}^{N} \frac{Z_i}{\pi_i} l_{ik} y_k = 0$$

$$\implies w_k = \sum_{i=1}^{N} \frac{Z_i}{\pi_i} l_{ik} \frac{1}{L_k} = \frac{\sum_{i=1}^{N} \frac{Z_i}{\pi_i} l_{ik}}{\sum_{i=1}^{N} l_{ik}}$$

The stuent is in S^B iff the student is linoked to one of the sample units in the S^A. i.e.

$$k \in S^B \iff \sum_{i \in S^A} l_{ik} > 0$$

$$k \notin S^B l_{ik} = 0 \; \forall i \in S^A$$

c.

If $L_k = 1$ we have that

$$\widehat{t}_y = \sum_{k \in S^B} \sum_{i=1}^{N} \frac{l_{ik} Z_i}{\pi_i} y_k = \sum_{i=1}^{N} \frac{Z_i}{\pi_i} \sum_{k \in S^B} l_{ik} y_k$$

Also, we obtain that each element k belongs to exactly one unit i. In this case, we can view our units in the same way we have viewed PSUs.

It follows that $\sum_{k \in S^B} l_{ik} y_k$ is also the total of the values in a given PSU i. Therefore, it is justified to write

$$\sum_{k \in S^B} l_{ik} y_k = t_i$$

$$\widehat{t}_y = \sum_{i=1}^{N} \frac{t_i}{\pi_i} Z_i = \widehat{t}_{HT}$$

We have the weight of each element as

$$w_k = \sum_{i=1}^{N} \frac{Z_i}{\pi_i} l_{ik} = \frac{Z_{ik}}{\pi_{ik}}$$

Where the indicator function for unit ik

$$Z_{ik} := \mathbf{1}_{ik}(jk)$$

and the inclusion probability

$$\pi_k = P(ik \text{ is in the sample})$$

Since unit k is only associated with a unique unit ik, it follows that

$$l_{jk} = 0$$

For all other j in U^A

d.

Remark that we are conducting SRS, we have $\pi_i = \frac{2}{3}$ $\forall$ unit i. There are three possible cases,

$$S_1 = \{1, 2\} \quad S_2 = \{1, 3\} \quad S_3 = 2, 3$$

Furthermore, we have $L_k = 2$ for $k = 1, 2$

Under the first case, we obtain that

$$\widehat{t_y} = \sum_{k=1}^{2}(.5)\sum_{i=1}^{3}(1.5)l_{ik}y_k = .75(l_{11} + l_{12} + l_{21} + l_{22})(y_1 + y_2) = 10.5$$

The second case:

$$\widehat{t_y} = .75 * (l_{21} + l_{31} + l_{22} + l_{32})(y_1 + y_2) = 7.5$$

and the third case:

$$\widehat{t_y} = .75 * (l_{11} + l_{31} + l_{12} + l_{32})(y_1 + y_2) = 12$$

We also remark that $\widehat{t_y}$ is unbiased.

We have

$$\widehat{t_y} = (.75)\sum_{k=1}^{2}\sum_{i=1}^{3} Z_i l_{ik} y_k$$

$$\implies E[\widehat{t_y}] = (.75)\sum_{k=1}^{2}\sum_{i=1}^{3} E[Z_i] l_{ik} y_k = \frac{3}{4}\sum_{k=1}^{2}\sum_{i=1}^{3}\frac{2}{3} l_{ik} y_k = \frac{1}{2}\sum_{k=1}^{2}\sum_{i=1}^{3} l_{ik} y_k$$

$$\frac{1}{2}(l_{11} + l_{31} + l_{21} + l_{22} + l_{12} + l_{32})(y_1 + y_2) = (.5)(4 + 4 + 6 + 6) = 10$$

From the formulae above (in part b), we are able to compute the variance

$$\pi_{ij} = \pi_{j|i}\pi_i = \frac{1}{3}$$

with u_i's (denoted as a 3-size sequence)

$$(u_i)_i = (2, 5, 3)$$

$$Var(\widehat{t_y}) = 19 - 15.5 = 3.5$$

e.

Below are the estimated values for the population total, SRSWR variance with CI's:

```r
library(tidyverse)
library(survey)
library(srvyr)
library(pps)
library(sampling)
library(data.table)
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## The following object is masked from 'package:purrr':
##
##     transpose
```

```r
wtshare <- read_csv('wtshare.csv')
```

```
## Parsed with column specification:
## cols(
##   Adult = col_double(),
##   child = col_double(),
##   preschool = col_double(),
##   numadults = col_double()
## )
```

```r
prsc <- as.vector(wtshare$preschool)
Lk <- 1/(as.vector(wtshare$numadults)+1)
lk <- as.vector(wtshare$child)

tHat <- prsc * Lk * lk

tHaty <- 400*sum(tHat)

#Estimated total:
print("Estimated total")
```

```
## [1] "Estimated total"
```

```r
tHaty
```

```
## [1] 7200
```

```r
setDT(wtshare)
wtshareSub <- wtshare[ , .(Total_Children = sum(preschool/(numadults+1))), by=.(Adult)]

u_i <- wtshareSub$Total_Children

SigmaSq <- ((40000*u_i)-tHaty)%*%((40000*u_i)-tHaty)/(99*100)


## Estimated Variance
print("Estimated Variance")
```

```
## [1] "Estimated Variance"
```

```r
SigmaSq
```

```
##          [,1]
## [1,] 1900606
```

```r
Sigma <- sqrt(SigmaSq)

L <- tHaty - (1.96*Sigma)
R <- tHaty + (1.96*Sigma)

print("95% confidence interval")
```

```
## [1] "95% confidence interval"
```

```r
c(L, R)
```

```
## [1] 4497.896 9902.104
```

6.23

a.

Provided values suggest

$$
\begin{aligned}
\pi_1 = \pi_{12} + \pi_{13} + \pi_{14} = .5 \\
\pi_2 = \pi_{21} + \pi_{23} + \pi_{24} = .25 \\
\pi_3 = \pi_{31} + \pi_{32} + \pi_{34} = .5 \\
\pi_4 = \pi_{41} + \pi_{42} + \pi_{43} = .75
\end{aligned}
$$

Below is the variance under w/o replacement (unequal prob)

```r
probs <- c(0.5, 0.25, 0.5, 0.75)
ts <- c(-5, 6, 0, -1)
beta1 <- probs*(1-probs)
beta2 <- (ts^2)/(probs^2)
vpt1 <- sum(c(beta1 * beta2))
probs2 <- c(0.004, 0.123, 0.373, 0.004, 0.123, 0.123, 0.123, 0.123, 0.254, 0.373,
0.123, 0.254)
probspairs <- c(0.5 * 0.25, 0.5 * 0.5, 0.5 * 0.75, 0.25 * 0.5,
0.25 * 0.5, 0.25 * 0.75, 0.5 * 0.5, 0.5 * 0.25, 0.5 * 0.75, 0.75 * 0.5,
0.75 * 0.25, 0.75 * 0.5)
tspairs <- c(-5 * 6, -5 * 0, -5 * -1, 6 * -5, 0, -6, 0, 0, 0, 5, -6, 0)
beta3 <- probs2-probspairs
beta4 <- tspairs/probspairs
vpt2 <- sum(c(beta3 * beta4))
VWOR <- vpt1 + vpt2
## Variance under an unequal probability sample:
VWOR
```

```
## [1] 195.488
```

```r
psys <- (1/2) * probs
trat <- ts/psys
beta5 <- trat^2
beta6 <- psys * beta5
```

```r
VWR <- (1/2) * sum(c(beta6))
## Variance under a with-replacement sample
VWR
```

```
## [1] 195.3333
```

SRSWR shows a lower variance, but by a very small margin; the bigger advantage comes from the fact that we may save the sampling cost.

b.

We know that $\psi_i = \pi_i/n$ and using (6.8) it follows that

$$n^{-1}\sum_{i=1}^{N}\psi_i\left(\frac{t_i}{\psi_i}-t\right)^2 = n^{-2}\sum_{i=1}^{N}\pi_i\left(\frac{nt_i}{\pi_i}-t\right)^2 = \sum_{i=1}^{N}\pi_i\left(\frac{nt_i}{\pi_i}-\frac{t}{n}\right)^2 =$$

$$\sum_{i=1}^{N}\frac{t_i^2}{\pi_i}+\sum_{i=1}^{N}\frac{t^2}{n^2}\pi_i-\sum_{i=1}^{N}\frac{t_i}{\pi_i}\frac{t}{n}\pi_i = \sum_{i=1}^{N}\frac{t_i^2}{\pi_i}+\frac{t^2}{n}-2\frac{t^2}{n} = \sum_{i=1}^{N}\frac{t_i^2}{\pi_i}-\frac{t^2}{n}$$

Working from RHS:

$$\sum_{i=1}^{N}\sum_{k=1}^{N}\pi_i\pi_k(\frac{t_i}{\pi_i}-\frac{t_k}{\pi_k})^2 = \sum_{i=1}^{N}\sum_{k=1}^{N}\pi_i\pi_k(\frac{t_i^2}{\pi_i^2}+\frac{t_k^2}{\pi_k^2}-2\frac{t_it_k}{\pi_i\pi_k}) =$$

$$\sum_{i=1}^{N}\sum_{k=1}^{N}\frac{t_i^2}{\pi_i}\pi_k+\sum_{i=1}^{N}\sum_{k=1}^{N}\frac{t_k^2}{\pi_k}\pi_i-2\sum_{i=1}^{N}t_i\sum_{k=1}^{N}t_k =$$

$$\sum_{i=1}^{N}\frac{t_i^2}{pi_i}\sum_{k=1}^{N}\pi_k+\sum_{i=1}^{N}\frac{t_k^2}{\pi_k}\sum_{k=1}^{N}\pi_i-2t^2 = n\left(\sum_{i=1}^{N}\frac{t_i^2}{\pi_i}+\sum_{k=1}^{N}\frac{t_k^2}{\pi_k}\right)-2t^2 = 2n\left(\sum_{k=1}^{N}\frac{t_i^2}{\pi_i}\right)-2t^2$$

$$\implies \frac{1}{2n}\sum_{i=1}^{N}\sum_{k=1}^{N}\pi_i\pi_k(\frac{t_i}{\pi_i}-\frac{t_k}{\pi_k})^2 = \sum_{i=1}^{N}\frac{t_i^2}{\pi_i}-\frac{t^2}{n}$$

Remarking both sides yields

$$Var(\widehat{t}_\psi) = \frac{1}{2n}\sum_{i=1}^{N}\sum_{k=1}^{N}\pi_i\pi_k(\frac{t_i}{\pi_i}-\frac{t_k}{\pi_k})^2 = \frac{1}{2n}\sum_{i=1}^{N}\sum_{k\neq i}^{N}\pi_i\pi_k(\frac{t_i}{\pi_i}-\frac{t_k}{\pi_k})^2$$

c.

Suppose that $\forall i,k\ \pi_{ik} \geq \frac{n-1}{n}\pi_i\pi_k$. Remark from the previous part as well.

by (6.21) we have

$$V(\widehat{t}_{HT}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{k\neq i}^{N}(\pi_i\pi_k-\pi_{ik})\left(\frac{t_i}{\pi_i}-\frac{t_k}{\pi_k}\right)^2 \leq \frac{1}{2}\sum_{i=1}^{N}\sum_{k\neq i}^{N}(\pi_i\pi_k-\frac{n-1}{n}\pi_i\pi_k)\left(\frac{t_i}{\pi_i}-\frac{t_k}{\pi_k}\right)^2 =$$

$$\frac{1}{2}\sum_{i=1}^{N}\sum_{k\neq i}^{N}(\frac{1}{n}\pi_i\pi_k)\left(\frac{t_i}{\pi_i}-\frac{t_k}{\pi_k}\right)^2=\frac{1}{2n}\sum_{i=1}^{N}\sum_{k\neq i}^{N}(\pi_i\pi_k)\left(\frac{t_i}{\pi_i}-\frac{t_k}{\pi_k}\right)^2=V(\widehat{t}_\psi)$$

In conclusion, we have shown

$$V(\widehat{t}_{HT})\geq V(\widehat{t}_\psi)$$

d. We denote for a given i

$$k_i=\min_k\frac{\pi_{ik}}{\pi_k}=\frac{\pi_{ik_i}}{\pi_{k_i}}=(n-1)\frac{\pi_i}{n}$$

$$\implies\sum_{i=1}^{N}\frac{\pi_{ik_i}}{\pi_{k_i}}\geq\frac{n-1}{n}\sum_{i=1}^{N}\pi_i=n-1$$

Meeting Gabler's condition.

e.

$$V(\widehat{t}_\psi)-V(\widehat{t}_{HT})=\frac{1}{2n}\sum_{i=1}^{N}\sum_{k\neq i}^{N}((n-1)\pi_i\pi_k+n\pi_{ik})\left(\frac{t_i^2}{\pi_i^2}+\frac{t_k^2}{\pi_k^2}-2\frac{t_it_k}{\pi_i\pi_k}\right)=$$

and we distribute the sum.

$$\frac{1}{2n}\sum_{i=1}^{N}\sum_{k\neq i}^{N}t_it_k(2(n-1)-2n\frac{\pi_{ik}}{\pi_i\pi_k})=\sum_{i=1}^{N}\sum_{k\neq i}^{N}t_it_k(\frac{n-1}{n}-\frac{\pi_{ik}}{\pi_i\pi_k})\geq 0$$

It suggests that the matrix $A=[(\frac{n-1}{n}-\frac{\pi_{ik}}{\pi_i\pi_k})_{ik}]=[a_{ik}]$ is positive semidefinite, implying its principal two by two blocks would have positive determinant.

Implying that

$$a_{ii}a_{kk}-a_{ik}a_{ki}\geq 0\implies(\frac{n-1}{n})^2\geq(\frac{n-1}{n}-\frac{\pi_{ik}}{\pi_i\pi_k})^2$$

$$\implies(\frac{n-1}{n})\geq(\frac{\pi_{ik}}{\pi_i\pi_k}-\frac{n-1}{n})\implies 2(\frac{n-1}{n})\geq\frac{\pi_{ik}}{\pi_i\pi_k}$$

$$\implies(\frac{n-1}{n}\pi_i\pi_k)\geq\pi_{ik}$$

6.45 IPUMS exercise

a

Select an unequal probability sampleof10 PSUs where $\psi_i\propto M_i$ where M_i would be the number of persons. Take a subsample of 20 persons in each PSU

b

Using the sample selected, estimate the population mean and total of inctot. Give the corresponding standard errors along with the estimates.

```
library(tidyverse)
library(survey)
library(srvyr)
library(pps)
library(sampling)
```

```
ipums <- read_csv('ipums.csv')
```

```
## Parsed with column specification:
## cols(
##   Stratum = col_double(),
##   Psu = col_double(),
##   Ssu = col_double(),
##   Inctot = col_double(),
##   Age = col_double(),
##   Sex = col_double(),
##   Race = col_double(),
##   Hispanic = col_double(),
##   Marstat = col_double(),
##   Ownershg = col_double(),
##   Yrsusa = col_double(),
##   School = col_double(),
##   Educrec = col_double(),
##   Labforce = col_double(),
##   Occ = col_double(),
##   Classwk = col_double(),
##   VetStat = col_double()
## )
```

```
head(ipums)
```

```
## # A tibble: 6 x 17
##   Stratum   Psu   Ssu Inctot   Age   Sex  Race Hispanic Marstat Ownershg
##     <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>    <dbl>   <dbl>    <dbl>
## 1       1     1     1   4105    18     1     2        0       5        0
## 2       1     1     2   7795    20     1     1        0       5        2
## 3       1     1     3  16985    24     1     1        0       1        1
## 4       1     1     4   7045    21     1     1        0       1        2
## 5       1     1     5   2955    23     1     1        0       5        2
## 6       1     1     6      0    17     1     1        0       5        1
## # ... with 7 more variables: Yrsusa <dbl>, School <dbl>, Educrec <dbl>,
## #   Labforce <dbl>, Occ <dbl>, Classwk <dbl>, VetStat <dbl>
```

```
dim(ipums)
```

```
## [1] 53461    17
```

```
#ipums_complete = ipums %>% filter(!is.na(Inctot))
#Verify that none of the data is missing
#dim(ipums_complete)
#no missing data
head(ipums)
```

```
## # A tibble: 6 x 17
##   Stratum   Psu   Ssu Inctot   Age   Sex  Race Hispanic Marstat Ownershg
##     <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>    <dbl>   <dbl>    <dbl>
## 1       1     1     1   4105    18     1     2        0       5        0
## 2       1     1     2   7795    20     1     1        0       5        2
## 3       1     1     3  16985    24     1     1        0       1        1
## 4       1     1     4   7045    21     1     1        0       1        2
## 5       1     1     5   2955    23     1     1        0       5        2
## 6       1     1     6      0    17     1     1        0       5        1
## # ... with 7 more variables: Yrsusa <dbl>, School <dbl>, Educrec <dbl>,
## #   Labforce <dbl>, Occ <dbl>, Classwk <dbl>, VetStat <dbl>
```

```
#We look for the number of distinct PSU's
num_PSU <- ipums %>% summarize(Num_PSU=n_distinct(Psu))
num_PSU
```

```
## # A tibble: 1 x 1
##   Num_PSU
##     <int>
## 1      90
```

```
ipums_tbl <- ipums %>% group_by(Psu) %>% summarise(Mi=n()) %>% ungroup() %>% mutate(N=n())
ipums_tbl %>% head(., n=10)
```

```
## # A tibble: 10 x 3
##      Psu    Mi     N
##    <dbl> <int> <int>
##  1     1   904    90
##  2     2  1082    90
##  3     3  1286    90
##  4     4  1094    90
##  5     5  1077    90
##  6     6  1020    90
##  7     7   951    90
##  8     8   928    90
##  9     9   985    90
## 10    10   974    90
```

```
print("Total number of schools that will be sampled")
```

```
## [1] "Total number of schools that will be sampled"
```

```
ipums_tbl %>% ungroup() %>% summarise(Ssutot = sum(Mi)) %>% head()
```

```
## # A tibble: 1 x 1
##   Ssutot
##    <int>
## 1  53461
```

```
set.seed(19970319)
## two-stage sampling without replacement
n=10

#Initialise

ipums_tbl = ipums_tbl %>% mutate(adj_size=Mi/sum(Mi))
```

```r
#Forcing the inclusion probabilities to be \leq 1
ipums_tbl = ipums_tbl %>% mutate(pi_k = inclusionprobabilities(Mi, n=n) )

##Create vector which indicates sampled PSU's

tille_sampled <- ipums_tbl %>% with(., UPtille(pi_k))

#Filters Mi table to include only the PSU's sampled
onestage_WOR <- ipums_tbl %>% filter(tille_sampled == 1)

#Grabs SSU's from sampled PSU's

onestage_WOR_sample <- inner_join(ipums, onestage_WOR, by="Psu")

#onestage_cluster_WOR_design <-  svydesign(id=~Psu, data=onestage_WOR_sample, fpc=~pi_k, pps="brewer")
onestage_WOR_sample
```

```
## # A tibble: 7,413 x 21
##    Stratum   Psu   Ssu Inctot   Age   Sex  Race Hispanic Marstat Ownershg
##      <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>    <dbl>   <dbl>    <dbl>
##  1       1     3     1   1520    23     1     1        0       1        2
##  2       1     3     2   2505    19     1     1        0       5        2
##  3       1     3     3   1325    18     1     2        0       5        2
##  4       1     3     4      0    22     1     1        0       5        1
##  5       1     3     5    255    20     1     2        0       1        2
##  6       1     3     6      0    18     1     2        0       5        2
##  7       1     3     7      0    17     1     2        0       5        1
##  8       1     3     8   1205    21     1     1        0       5        1
##  9       1     3     9      0    18     1     1        0       5        1
## 10       1     3    10    315    15     1     1        0       5        1
## # ... with 7,403 more rows, and 11 more variables: Yrsusa <dbl>,
## #   School <dbl>, Educrec <dbl>, Labforce <dbl>, Occ <dbl>, Classwk <dbl>,
## #   VetStat <dbl>, Mi <int>, N <int>, adj_size <dbl>, pi_k <dbl>
```

```r
set.seed(19970319)
#One stage done
##Stage 2
m=20
twostage_WOR_sample <- onestage_WOR_sample %>% group_by(Psu)  %>%  sample_n(size=m, replace=F) %>% ungr

#twostage_WOR_sample
twostage_cluster_WOR_design <- svydesign(id=~Psu+Ssu, data=twostage_WOR_sample, weight=~I(1/pi_k)+I(Mi/


print("Estimated population total of Inctot")
```

```
## [1] "Estimated population total of Inctot"
```

```r
svytotal(~Inctot, twostage_cluster_WOR_design)
```

```
##            total       SE
## Inctot 490020853 59263530
```

```r
print("Estimated population mean of Inctot")
```

```
## [1] "Estimated population mean of Inctot"
```

```r
svymean(~Inctot, twostage_cluster_WOR_design)
```

```
##        mean     SE
## Inctot 9166 1108.5
```