final

Take home Final

Name: Yi Chen, UNI: yc3356

Problem 1.

```
final_data <- read.csv('FinalPop.csv.xls')
y_bar_u <- mean(final_data$y)
s_2 <- var(final_data$y)
cat('the mean is :',y_bar_u,'\n')
```

```
## the mean is : 14.882
```

```
cat('the variance is:',s_2)
```

```
## the variance is: 0.9774571
```

a)

1. Clearly, SRS is design-based unbiased. In other words: $E(\bar{y} - \bar{y}_U) = 0$

Prove:

$$E(\bar{y} - \bar{y}_U) = E(\bar{y}) - \bar{y}_U = E(\frac{1}{n}\sum_{i:i \in S} y_i) - \bar{y}_U$$

$$= \frac{1}{n}\sum_{i=1}^{N} y_i E(z_i) - \bar{y}_U = \frac{1}{n}\frac{n}{N}\sum_{i=1}^{N} y_i - \bar{y}_U = 0$$

Here $z_i = 1$ if the i-th element is in the sample.

2.

$$E[(\bar{y} - \bar{y}_U)^2] = Bias(\bar{y})^2 + Var(\bar{y}) = 0 + \frac{S^2}{n}(1 - \frac{n}{N})$$

```
N = 50
n = 10
MSE <- (s_2 / n) * (1- n /N)
cat('the bais is 0 \n')
```

```
## the bais is 0
```

```
cat('the MSE is:',MSE)
```

```
## the MSE is: 0.07819657
```

b)

1. Clearly, stratification sampling is design-based unbiased. In other words: $E(\overline{y} - \overline{y}_U) = 0$

Prove:

$$E(\overline{y}_{str} - \overline{y}_U) = E(\overline{y}_{str}) - \overline{y}_U = E(\frac{\hat{t}}{N}) - \overline{y}_U = \frac{\sum_{h=1}^{H} t_h}{N} - \overline{y}_U = 0$$

2.

$$E[(\overline{y} - \overline{y}_U)^2] = Bias(\overline{y}_{str})^2 + Var(\overline{y}_{str}) = 0 + \frac{1}{N^2}\sum_{h=1}^{H} N_h^2 \frac{s_h^2}{n_h}(1 - \frac{n_h}{N_h})$$

```
N_h <- rep(10,5)
n_h <- rep(2,5)
v_h <- apply(matrix(final_data$y,nrow = 10),2,var)
MSE <-  sum((N_h/N)^2 * (v_h /n_h) * (1 - n_h/N_h))
cat('the bais is 0 \n')
```

```
## the bais is 0
```

```
cat('the MSE is:',MSE)
```

```
## the MSE is: 0.08310425
```

c)

$$E_M(\overline{y} - \overline{y}_U) = E_M(\frac{\sum_{i\in s} Y_i}{n}) - \frac{\sum_{i=1}^{N} Y_i}{N} = \frac{n\mu}{n} - \mu = 0$$

$$E_M[(\overline{y} - \overline{y}_U)^2] = E_M[(\frac{\sum_{i\in s} Y_i}{n} - \frac{\sum_{i=1}^{N} Y_i}{N})^2] = E_M[((\frac{1}{n} - \frac{1}{N})\sum_{i\in s} Y_i - \frac{1}{N}\sum_{i\in s^c} Y_i)^2]$$

$$= E_M[(\frac{1}{n} - \frac{1}{N})^2(\sum_{i\in s} Y_i - n\mu)^2) + (\sum_{i\in s^c} Y_i - (1 - \frac{n}{N})\mu)^2] = \frac{S^2}{n}(1 - \frac{n}{N})$$

```
sigma <- 1
MSE <- (1 - n/N) * (sigma / n)
cat('the bais is 0 \n')
```

```
## the bais is 0
```

```
cat('the MSE is:',MSE)
```

```
## the MSE is: 0.08
```

d)

1. Stratify Sampling is model based unbiased: $E(\bar{y} - \bar{y}_U) = 0$

Prove:

$$E(\bar{y}_{str} - \bar{y}_U) = E(\bar{y}_{str}) - \bar{y}_U = E(\frac{\hat{t}}{N}) - \bar{y}_U = \frac{\sum_{h=1}^{H} t_h}{N} - \bar{y}_U = 0$$

2.

$$E_M[(\bar{y} - \bar{y}_U)^2] = Bias_M(\bar{y}_{str})^2 + Var_M(\bar{y}_{str}) = \sum_{h=1}^{H} \frac{1}{N^2} \sum_{h=1}^{H} N_h^2 \frac{s_h^2}{n_h}(1 - \frac{n_h}{N_h})$$

```
sigma_h <- rep(1,5)
N_h <- rep(10,5)
n_h <- rep(2,5)
MSE <-  sum((N_h/N)^2 * (sigma_h^2 /n_h) * (1 - n_h/N_h))
cat('the bais is 0 \n')
```

```
## the bais is 0
```

```
cat('the MSE is:',MSE)
```

```
## the MSE is: 0.08
```

problem 2

```
library(SDaA)
Data <- counties[,c(2,3,5,17)]
t.x <- 255077536
```

a) ratio estimation

```
N <- 3141
n <- 100
y_bar <- mean(Data$veterans)
x_bar <- mean(Data$totpop)

B.hat <- y_bar/x_bar
t.hat.y <- B.hat*t.x
cat('the total number of veterans is estimated to be :', round(t.hat.y/1000000,2),'milli
on. \n')
```

```
## the total number of veterans is estimated to be : 26.36 million.
```

```
error <- Data$veterans-B.hat*Data$totpop
s2.e <- var(error)
v.B.hat <- s2.e/(n*x_bar^2) * (1-n/N)
v.t.hat.y <- t.x^2 * v.B.hat
SE <- sqrt(v.t.hat.y)
cat('the confidence interval is estimated to be : [', round((t.hat.y + c(-1.96,1.96) * S
E)/1000000,2)[1],',',round((t.hat.y + c(-1.96,1.96) * SE)/1000000,2)[2],'] million.')
```

```
## the confidence interval is estimated to be : [ 23.15 , 29.57 ] million.
```

b) regression estimation

```
x <- Data$totpop
y <- Data$veterans
xbar.U <- t.x / N
xbar <- mean(x); ybar <- mean(y); s.y <- sd(y);
B1.hat <- cor(x,y) * sd(y) / sd(x)
B0.hat <- ybar - B1.hat * xbar
ybar.hat.reg <- ybar + B1.hat * (xbar.U - xbar)
e <- y - B0.hat - B1.hat * x
SE.ybar.reg <- sd(e) / sqrt(n) * sqrt(1 - n/N)
t_reg <- N * ybar.hat.reg;
sd_t_reg <- N * SE.ybar.reg;
cat('the total number of veterans is estimated to be :', round(t_reg/1000000,2),'millio
n. \n')
```

```
## the total number of veterans is estimated to be : 27.88 million.
```

```
cat('the confidence interval is estimated to be : [', round((t_reg + c(-1.96,1.96) * sd_
t_reg)/1000000,2)[1],',',round((t_reg + c(-1.96,1.96) * sd_t_reg)/1000000,2)[2],'] milli
on.')
```

```
## the confidence interval is estimated to be : [ 25.64 , 30.12 ] million.
```

c) regression estimation

```
Sxx <- (n-1) * var(x)
m1 <- lm(y~x)
sigma.hat <- sigma(m1)
SE_M.yar.reg <- sigma.hat * sqrt(1/n + (xbar.U - xbar)^2 / Sxx)
SE_M.y.total.reg <- N * SE_M.yar.reg
cat('design based SE:',sd_t_reg,'\n')
```

```
## design based SE: 1142010
```

```
cat('model based SE:',SE_M.y.total.reg)
```

```
## model based SE: 1169519
```

Thus the model based SE is little bit bigger.

d)

$$V_M[\hat{T}_y - T] = (1 - \frac{\sum_{i \in s} x_i}{t_x})\frac{\sigma^2 t_x^2}{\sum_{i \in s} x_i}$$

```
m1 <- lm(y ~ 0 + x, weights=1/x)
sigma.hat <- sigma(m1)
t_x <- 255077536
SE_M.t_yr <- sigma.hat * t_x / sqrt(n*xbar) * sqrt(1 - (n*xbar)/t_x)
SE_M.t_yr
```

```
## [1] 457516.8
```

The standard deviation in part (a) for the design approach is 3623183. The model approach standard deviation is much bigger.

problem 3

a)

```
n1 <- 100
n2 <- 80
m <- 18
N_hat <- n1 * n2 / m
V_N <- ( n1 * n2 / m )^2 * ((n2 -m) / (m*(n2-1)))
SE <- sqrt(V_N)
N_hat ; SE
```

```
## [1] 444.4444
```

```
## [1] 92.80332
```

The maximum likelihood esitmation is 444 and the standard error is 93.

biased

b)

Based on the information (data) we observed, our best guess about the number of fish in the pond is 444. But this is an unknow value which we cannot guarantee. we can say 92 (standard devaition) represents the amount by which our estimation is likely to be off.

c)

Observed	second sample (Y)	second sample (N)
first sample (Y)	$m = 18$	$n_1 - m = 82$
first sample (N)	$n_2 - m = 62$	$ N + m - n_1 - n_2 = N -162 $

Expected	second sample (Y)	second sample (N)
first sample (Y)	$n_1 n_2 / N = 8000/N$	$(N - n_1) n_2 / N = 80 - 8000/N$
first sample (N)	$n_1 (N - n_2)/N = 100 - 8000/N$	$(N-n_1)(N- n_2)/N $

$$\chi_2 = \sum \frac{(observed - expected)^2}{expected}$$

```
conf.level = 0.9
alpha <- 1 - conf.level
x11 <- m; x12 <- n1 - m; x21 <- n2 - m;
x22.hat <- round(N_hat - x11 - x12 - x21)
X <- matrix(c(x11, x21, x12, x22.hat),2,2)
Reject <- FALSE
u <- x22.hat;
while(!Reject)
{
  X <- matrix(c(x11,x21,x12,u),2,2)
  Reject <- chisq.test(X, correct=F)$p.value < alpha
  u <- u - 1
}

u <- u + 2
N.lower <- x11 + x12 + x21 + u; N.lower
```

```
## [1] 336
```

```
u <- x22.hat
Reject <- FALSE;
while(!Reject)
{
  X <- matrix(c(x11,x21,x12,u),2,2)
  Reject <- chisq.test(X, correct=F)$p.value < alpha
  u <- u + 1
}

u <- u - 2
N.upper <- x11 + x12 + x21 + u; N.upper
```

```
## [1] 618
```

Thus, we can say that the confidence interval should be [336,618]

4

a)

```
state_pop <- read.csv('statepop.csv.xls')
county <- read.csv('counties.csv')
M.0 <- sum(state_pop$popn)
M.i <- state_pop$popn
pis.i <- M.i / M.0
t.i <- county$counties
index <- c()
for (i in county$state){
  index <- c(index,which(state_pop$state == i))
}
u.i <- t.i / pis.i[index]
t_hat <- mean(u.i)
n <- length(t.i)
v_t_hat <- (1/n) * (1/(n-1)) * sum((u.i - mean(u.i))^2)
se_t_hat <- sqrt(v_t_hat)
t_hat ; se_t_hat
```

```
## [1] 2353.318
```

```
## [1] 648.8684
```

Thus, total numebr of counties is estimated to be 2353 and the standard error is estimated to be 649.

b

$R^* = CA, CO, CT, MA, MO, NJ, TN, VA, WI$

```
state <- c('CA', 'CO', 'CT', 'MA', 'MO', 'NJ', 'TN', 'VA', 'WI')
psi_weight <- c()
psi_weight <- c(psi_weight,pis.i[which(state_pop$state == 'California')])
psi_weight <- c(psi_weight,pis.i[which(state_pop$state == 'Colorado')])
psi_weight <- c(psi_weight,pis.i[which(state_pop$state == 'Connecticut')])
psi_weight <- c(psi_weight,pis.i[which(state_pop$state == 'Massachusetts')])
psi_weight <- c(psi_weight,pis.i[which(state_pop$state == 'Missouri')])
psi_weight <- c(psi_weight,pis.i[which(state_pop$state == 'New Jersey')])
psi_weight <- c(psi_weight,pis.i[which(state_pop$state == 'Tennessee')])
psi_weight <- c(psi_weight,pis.i[which(state_pop$state == 'Virginia')])
psi_weight <- c(psi_weight,pis.i[which(state_pop$state == 'Wisconsin')])
q.i <- c()
q.i <- c(q.i,sum(county$state == 'California'))
q.i <- c(q.i,sum(county$state == 'Colorado'))
q.i <- c(q.i,sum(county$state == 'Connecticut'))
q.i <- c(q.i,sum(county$state == 'Massachusetts'))
q.i <- c(q.i,sum(county$state == 'Missouri'))
q.i <- c(q.i,sum(county$state == 'New Jersey'))
q.i <- c(q.i,sum(county$state == 'Tennessee'))
q.i <- c(q.i,sum(county$state == 'Virginia'))
q.i <- c(q.i,sum(county$state == 'Wisconsin'))

w.i <- 1 / (n * psi_weight)
result <- data.frame('state'=state, 'w.i'=w.i, 'Q.i'=q.i,'w.i*Q.i'=w.i*q.i)
result
```

```
##   state       w.i Q.i  w.i.Q.i
## 1    CA 0.6880136   3 2.064041
## 2    CO 6.1351862   1 6.135186
## 3    CT 6.4823649   1 6.482365
## 4    MA 3.5470462   1 3.547046
## 5    MO 4.0950832   1 4.095083
## 6    NJ 2.7181227   2 5.436245
## 7    TN 4.2299149   1 4.229915
## 8    VA 3.3241832   1 3.324183
## 9    WI 4.2575319   1 4.257532
```

```
sum(w.i * q.i)
```

```
## [1] 39.5716
```

Think in every sample: $Q_i = 0$ if it is not in the sample. But $E(Q_i) > 0$ for every state.

$E(w_i) = \frac{1}{n\psi_i}$ and $E(Q_i) = n\psi_i$

$E(W_i Q_i) = 1$ and $E(\sum_{i=1}^{N} W_i Q_i) = N$

problem 5

```
screentime <- read.csv("ScreenTime.csv")
N <- 2000
n <- 200
n_R <- n*0.75
n_M <- n*0.25
v <- 10

group <- split(screentime$Minutes, screentime$Group)
ybar_R <- sapply(group, mean)[[1]]
ybar_M <- sapply(group, mean)[[2]]

ybar.hat <- (n_R/n)*ybar_R + (n_M/n)*ybar_M
ybar.hat
```

```
## [1] 195.27
```

```
s2_R <- sapply(group, var)[[1]]
s2_M <- sapply(group, var)[[2]]

term2 <- (n_R/n)*(ybar_R-ybar.hat)^2 + (n_M/n)*(ybar_M-ybar.hat)^2
v.ybar.hat <- (n_R-1)/(n-1)*(s2_R/n) + (n_M-1)/(n-1)*(s2_M/(v*n)) + 1/(n-1)*term2
SE <- sqrt(v.ybar.hat)
SE
```

```
## [1] 7.420013
```

```
screentime <- read.csv('ScreenTime.csv')
n_r <- sum(screentime$Group == 1)
n_m <- nrow(screentime) - n_r
v <- 10 / n_m

N <- 2000
n <- nrow(screentime)
group <- split(screentime$Minutes, screentime$Group)
y_bar_r <- sapply(group, mean)[[1]]
y_bar_m <- sapply(group, mean)[[2]]
s_r <- sapply(group, sd)[[1]]
s_m <- sapply(group, sd)[[2]]
y_bar_hat <- (n_r / n) * y_bar_r + (n_m / n) * y_bar_m
y_bar_hat
```

```
## [1] 195.27
```

```
v_ybar_hat <- (n_r-1)/(n-1)*(s_r^2/n) + (n_m-1)/(n-1)*(s_m^2/(v*n)) + 1/(n-1)*((n_r/n)*
(y_bar_r-y_bar_hat)^2 + (n_m/n)*(y_bar_m-y_bar_hat)^2)
se_ybar_hat <- sqrt(v_ybar_hat)
se_ybar_hat
```

```
## [1] 13.66384
```

```
y_bar_hat + c(-1.96,1.96) * se_ybar_hat
```

```
## [1] 168.4889 222.0511
```

The 95% confidence interval is [168.4889，222.0511]