# STA 522 HW3 (Chapter 2 & Horvitz-Thompson)

*Daniel Truver*

*1/25/2018*

**(1) SRS Design Precision**

From Lohr, we know the formula for variance to be

$$Var(\bar{y}) = \frac{S^2}{n}\left(1 - \frac{n}{N}\right)$$

If we are very lucky, or very generous with our assumptions, and $S^2$ is the same for all given populations, then we have a straightforward means of comparing variances.

$$\begin{aligned}
Var(\bar{y}_1) &= \frac{S^2}{400}\left(1 - \frac{400}{4000}\right) = \frac{9S^2}{4000} \\
Var(\bar{y}_2) &= \frac{S^2}{30}\left(1 - \frac{30}{300}\right) = \frac{3S^2}{100} \\
Var(\bar{y}_3) &= \frac{S^2}{3000}\left(1 - \frac{3000}{300{,}000{,}000}\right) = \frac{0.99999S^2}{3000}
\end{aligned}$$

Of these, the third sample has the lowest variance and therefore the highest precision. Even though the proportion of the population sampled is lowest here, the pure size of the sample is really the deciding factor. This result aligns with our intuition about sampling.

**(2) SRSWR**

(a) The event that we select $k$ at least one time is the complement of the event that we never select $k$.

$$\pi_k = Pr(I_k = 1) = 1 - Pr(I_k = 0) = 1 - Pr(k \text{ is never selected}) = 1 - (1 - p_k)^n$$

(b)

$$\hat{t}_y = \sum_{i \in S} y_k/\pi_i = \sum_{i \in S} \frac{y_i}{1 - (1 - p_i)^n}$$

(c) We unfortunately want to find the expression for $Pr(I_k = 1, I_l = 1); k \neq l$. This is the complement of ($k$ never selected) $\cup$ ($l$ never selected).

$$\begin{aligned}
\pi_{ik} &= Pr(I_k = 1, I_l = 1) \\
&= 1 - (Pr(k \text{ unselected}) + Pr(l \text{ unselected}) - Pr(k \text{ unselected} \cap l \text{ unselected})) \\
&= 1 - (Pr(k \text{ unselected}) + Pr(l \text{ unselected}) - (1 - Pr(k \text{ selected} \cup l \text{ selected}))^n) \\
&= 1 - (1 - p_k)^n - (1 - p_l)^n + (1 - (p_k + p_l))^n
\end{aligned}$$

as was to be shown.

(d) Oh dear, parantheses hell closes in.

$$\begin{aligned}
\hat{Var}(\hat{t_y}) &= \sum_{i\in S}\left(\frac{y_i}{\pi_i}\right)^2(1-\pi_i) + 2\sum_{i<j\in S}\frac{y_i y_j}{\pi_i\pi_j\pi_{ij}}(\pi_{ij}-\pi_i\pi_j) \\
&= \sum_{i\in S}\left(\frac{y_i}{1-(1-p_i)^n}\right)^2(1-p_i)^n \\
&+ 2\sum_{i<j\in S}\frac{y_i y_j}{[1-(1-p_i)^n]\cdot[1-(1-p_j)^n]\cdot[1-(1-p_k)^n-(1-p_l)^n+(1-(p_k+p_l))^n]} \\
&\cdot\Big(1-(1-p_k)^n-(1-p_l)^n+(1-(p_k+p_l)^n-(1-(1-p_i)^n)\cdot(1-(1-p_j)^n)\Big)
\end{aligned}$$

**(3) SRSWR (incompetent friend version)**

(a/b) Following the derivation, we will proceed to beat our friend about the head and shoulders with this rolled up paper.

$$\begin{aligned}
E(\tilde{t_y}) &= E\left(\sum_{k\in S} y_k/p_k\right) \\
&= E\left(\sum_{k=1}^{N}\frac{y_k}{p_k}I_k\right) \\
&= \sum_{k=1}^{N}\frac{y_k}{p_k}E(I_k) \\
&= \sum_{k=1}^{N}\frac{y_k}{p_k}(1-(1-p_k)^n) \\
&\Longrightarrow \\
\text{Bias}(\tilde{t_y}) &= \left(\sum_{k=1}^{N}\frac{y_k}{p_k}(1-(1-p_k)^n)\right) - t_y
\end{aligned}$$

The beatings shall continue until estimators improve.

**(4) Simulation**

(a) Copy and pasting... I mean... setting up the simulation.

```
#### Simulation code for Methods 3


T = 10000 #this is the number of simulation runs.
output = matrix(nrow = T, ncol = 3)   #this is where we store the output.

N = 100000  #set to desired population size
n = 25000 #set to desired sample size

#make population data  -- let's use a binary outcome with p = 0.2
popdata = rbinom(N, 1, p = 0.2)

run.cores = parallel::detectCores()/2
set.seed(2018)
```

```
res = parallel::mclapply(1:T,
                         function(j){
                           sampdata = popdata[sample(1:N, n, replace = FALSE)]
                           ybar = mean(sampdata)
                           varWithFPC = (1 - n/N)*var(sampdata)/n
                           varNoFPC = var(sampdata)/n
                           return(list("ybar" = ybar,
                                       "varWithFPC" = varWithFPC,
                                       "varNoFPC" = varNoFPC))
                         },
                         mc.cores = run.cores)
output = data.frame(t(matrix(unlist(res), nrow = 3, ncol = T)))
means = apply(output, 2, mean)
knitr::kable(data.frame(means[1], var(output[,1]), means[2], means[3]),
             col.names = c("average y_bar", "var(y_bar)",
                           "average corrected variance", "average vanilla variance"),
             caption = "Comparison of corrected and uncorrected variance (n = 25,000)")
```

Table 1: Comparison of corrected and uncorrected variance (n = 25,000)

| | average y_bar | var(y_bar) | average corrected variance | average vanilla variance |
|---|---|---|---|---|
| X1 | 0.2012239 | 4.8e-06 | 4.8e-06 | 6.4e-06 |

(b) We see from the results of part (a) that the estimated variance with finite population correction is closer on average to the variance of the sample mean than is the uncorrected estimated variance. We conclude that the finite population correction is important to obtain an accurate estimate for the variance of the sample mean, given these sample and population sizes.

(c) Copy. Paste. Profit.

```
T = 10000 #this is the number of simulation runs.
output = matrix(nrow = T, ncol = 3)

N = 100000  #set to desired population size
n = 500 #set to desired sample size

#make population data  -- let's use a binary outcome with p = 0.2
popdata = rbinom(N, 1, p = 0.2)

run.cores = parallel::detectCores()/2
set.seed(2018)
res = parallel::mclapply(1:T,
                         function(j){
                           sampdata = popdata[sample(1:N, n, replace = FALSE)]
                           ybar = mean(sampdata)
                           varWithFPC = (1 - n/N)*var(sampdata)/n
                           varNoFPC = var(sampdata)/n
                           return(list("ybar" = ybar,
                                       "varWithFPC" = varWithFPC,
                                       "varNoFPC" = varNoFPC))
                         },
                         mc.cores = run.cores)
output = data.frame(t(matrix(unlist(res), nrow = 3, ncol = T)))
```

```
means = apply(output, 2, mean)
knitr::kable(data.frame(means[1], var(output[,1]), means[2], means[3]),
             col.names = c("average y_bar", "var(y_bar)",
                           "average corrected variance", "average vanilla variance"),
             caption = "Comparison of corrected and uncorrected variance (n = 500)")
```

Table 2: Comparison of corrected and uncorrected variance (n = 500)

| | average y_bar | var(y_bar) | average corrected variance | average vanilla variance |
|---|---|---|---|---|
| X1 | 0.1993396 | 0.0003242 | 0.0003176 | 0.0003192 |

Using the finite population correction with these sample and population sizes does not seem to be necessary. This empirical result aligns with what we would expect from intuition. The finite population correction is only $1-n/N = $ 0.995, not much of a change.