Comparing One-stage cluster sampling to SRS

Week 7 (5.2.2)

Stat 260, St. Clair

When is a one-stage cluster sample more precise than SRS?

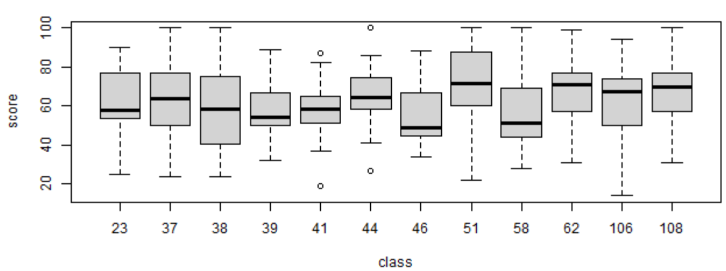
When does

$$SE(\hat{t}_{cluster}) \overset{???}{<} SE(\hat{t}_{SRS})$$

answer: It depends on the measurement's **Analysis of Variance** (ANOVA)

Lohr Examples 5.6: design effect

```
> svymean(~score, alg_design, deff = TRUE)
         mean      SE  DEff
score 62.5686  1.4916 2.245
> boxplot(score ~ class, data = algebra)
```



Population ANOVA

Let y_{ij} be your measurement of unit j in cluster i

ANOVA breaks the **total** sum of squares of y into **between cluster** and **within cluster** variation:

$$SST = SSB + SSW$$

For now, **assume that cluster sizes are equal**

$$M_i = M \text{ for all clusters } i = 1, \ldots, N$$

Population ANOVA

Source	df	Sum of Squares	Mean Square
Between	$N-1$	$SSB = \sum_{i=1}^{N} M(\bar{y}_{i\mathcal{U}} - \bar{y}_{\mathcal{U}})^2$	$MSB = \frac{SSB}{N-1}$
Within	$N(M-1)$	$SSW = \sum_{i=1}^{N}(M-1)S_i^2$	$MSW = \frac{SSW}{N(M-1)}$
total	$NM-1$	$SSTot = \sum_{i=1}^{N}\sum_{j=1}^{M}(y_{ij} - \bar{y}_{\mathcal{U}})^2$	$S^2 = \frac{SSTot}{NM-1}$

Variance: SRS

Equal cluster sizes: We've sampled nM **observation units** (SSU) out of $M_0 = NM$ possible units.

For a SRS of nM **observation units**, we can write the variance, SE^2, of $\hat{t}_{SRS}$ as

$$Var(\hat{t}_{SRS}) = (NM)^2 \left(1 - \frac{nM}{NM}\right) \frac{S^2}{nM}$$

where S is the SD of the measurements in the population.

Variance: One-stage cluster sample

Equal cluster sizes: Under this assumption the variance of $\hat{t}_{unb}$ is equal to

$$Var(\hat{t}_{unb}) = N^2 \left(1 - \frac{n}{N}\right) \frac{M \times MSB}{n}$$

Variance: SRS vs. Stratified sample

Equal cluster sizes: Under this assumption, the design effect for a one-stage cluster sample total estimate is

$$DEff(\hat{\bar{y}}_{unb}) = DEff(\hat{t}_{unb}) = \frac{Var(\hat{t}_{unb})}{Var(\hat{t}_{SRS})} = \frac{MSB}{S^2}$$

Variance: SRS vs. Stratified sample

Cluster sampling is more precise than an equal sized SRS when

$$MSB < S^2$$

$$\Rightarrow \text{ between cluster variation is small}$$

$$\Rightarrow \text{ measurements are heterogenous within clusters}$$

Measuring homogeneity within clusters

- **Intraclass correlation coefficient:** for equal sized clusters

$$ICC = 1 - \frac{M}{M-1}\frac{SSW}{SSTot} \quad \text{where } -\frac{1}{M-1} \leq ICC \leq 1$$

- **Adjusted R-squared:** can be used for unequal cluster sizes

$$R_a^2 = 1 - \frac{MSW}{S^2} \quad \text{where } 1 - \frac{NM-1}{N(M-1)} \leq R_a^2 \leq 1$$

- **For both:**
  - values near 1 indicate **homogeneous** (similar) responses **within** clusters
  - values near 0 indicate **heterogeneous** (dissimilar) responses **within** clusters

Design effect revisted

Equal cluster sizes: Under this assumption the DEff of $\hat{t}_{unb}$ is equal to

$$\begin{aligned} DEff(\hat{t}_{unb}) &= \frac{MSB}{S^2} \\ &= \frac{MN-1}{M(N-1)}(1+(M-1)ICC) \\ &= 1 + \frac{N(M-1)}{N-1}R_a^2 \end{aligned}$$

Design effect revisted

What is the design effect if

- N is big
- $M = 11$
- $R_a^2 = 0.5$

Big picture

- One-stage cluster sampling is "good" for precision if SSU within clusters have very **heterogeneous** responses
  - true whether or not cluster sizes are equal
- But often SSU within clusters have very **homogeneous** responses
  - clusters contain "similar" observation units
  - clusters defined for **cost-saving** reasons, not for precision

Post-Hoc comparison

Q: How do we compute the design effect with a **sample of data** from any one-stage cluster sample?

1. (Any cluster sizes) Use sampling weights to estimate $Var(\hat{t}_{srs})$

- This is what the `survey` package when you use `deff=TRUE`

Post-Hoc comparison

Q: How do we compute the design effect with a **sample of data** from any one-stage cluster sample?

2. Equal cluster sizes: Estimate population sum of square values from **sample** mean square values msw and msb:

$$\widehat{SSW} = N(M-1)msw \quad \widehat{SSB} = (N-1)msb$$

The estimated design effect is

$$\widehat{DEff}(\hat{t}_{unb}) = \frac{\widehat{MSB}}{\hat{S}^2} = \frac{msb}{(\widehat{SSW} + \widehat{SSB})/(NM-1)}$$

Estimating ICC and R_a^2

$\widehat{SSW} = N(M-1)msw, \quad \widehat{SSB} = (N-1)msb, \quad \widehat{SST} = \widehat{SSB} + \widehat{SSW}$

- Estimated ICC is

$$ICC = 1 - \frac{M}{M-1}\frac{\widehat{SSW}}{\widehat{SST}}$$

- Estimated R_a^2 is

$$\hat{R}_a^2 = 1 - \frac{msw}{\hat{S}^2}$$

Example - GPA

$N = 100, n = 5, M_i = 4, M_0 = 400$

	Suite 1	Suite 2	Suite 3	Suite 4	Suite 5
1	3.08	2.36	2.00	3.00	2.68
2	2.60	3.04	2.56	2.88	1.92
3	3.44	3.28	2.52	3.44	3.28
4	3.04	2.68	1.88	3.64	3.20
total	12.16	11.36	8.96	12.96	11.08

```
> dorm <- read.csv("http://math.carleton.edu/kstclair/data/Dorm_Clust
> dplyr::glimpse(dorm)
Rows: 20
Columns: 2
$ gpa  <dbl> 3.08, 2.60, 3.44, 3.04, 2.36, 3.04, 3.28, 2.68, 2.00, 2.
$ room <int> 1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5,
```


Example - GPA

$N = 100, n = 5, M_i = 4, M_0 = 400$

What is the design effect, ICC and R_a^2 for estimating mean GPA?

```
> dorm_lm <- lm(gpa ~ factor(room), data = dorm)
> anova(dorm_lm)
Analysis of Variance Table

Response: gpa
             Df Sum Sq Mean Sq F value  Pr(>F)
factor(room)  4 2.2557 0.56392  3.0476 0.05039 .
Residuals    15 2.7756 0.18504
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```


Lohr Examples 5.6: design effect of 2.245

What if cluster sizes are not equal?

```
> alg_lm <- lm(score ~ factor(class), data = algebra)
> anova(alg_lm)
Analysis of Variance Table

Response: score
               Df Sum Sq Mean Sq F value  Pr(>F)
factor(class)  11   7086  644.14  2.1184 0.01915 *
Residuals     287  87270  304.08
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> msb <- 644.14
> msw <- 304.08
> msb/var(algebra$score)   # rough DEff guess
[1] 2.03437
```

Lohr Examples 5.6: design effect of 2.245

What if cluster sizes are not equal?

```
> summary(alg_lm)$adj.r.squared  # rough R^2_a guess
[1] 0.03964506
> library(dplyr)
> algebra %>%
+   group_by(class) %>%
+   summarize(Mi = n()) %>%  # gets Mi values by class
+   summarize(mean(Mi))      # mean Mi per class
# A tibble: 1 x 1
  `mean(Mi)`
       <dbl>
1       24.9
> 1 +  187*(25-1)*.04/(187-1)  # rough DeFF guess based on R^2_a
[1] 1.965161
```