Cluster Sampling

- A **cluster sample** is a probability sample in which each sampling unit is a collection or a group of elements.
- It is useful when:

  (*i*) A list of elements of the population is not available but it is easy to obtain a list of clusters.

  (*ii*) The cost of obtaining observations increases as the distance that separates the elements.
- If only a **sample** of elements is taken from each selected cluster, the method is known as *two-stage* sampling.
- Often a hierarchy of clusters is used: First some large clusters are selected, next some smaller clusters are drawn within the selected large clusters, and so on until finally elements are selected within the final-stage clusters.

- EXAMPLE: In a survey of students from a city, we first select a sample of schools, then we select a sample of classrooms within the selected schools, and finally we select a sample of students within the selected classes.
- This general method is known as *multistage sampling*, although it is also sometimes loosely described as cluster sampling.
- Although strata and clusters are both groupings of elements, they serve for entirely **different** sampling purposes.
- Since **strata** are all represented in the sample, it is advantageous if they are internally **homogeneous** in the survey variables.
- As only a sample of clusters are sampled, the ones selected need to represent the ones unselected; this is best done when the clusters are as internally **heterogeneous** in the survey variables *as possible*.

- Except in special circumstances, cluster sampling leads to a loss in precision compared with an *SRS* of the same size but it can be justified by a better *economy*.

  We can use the **following notations**:

  N the number of clusters in the population.

  n the number of clusters *selected* in a simple random sample.

  m_i the number of elements in cluster i, $i = 1, \ldots, N$.

  $\bar{m} = \frac{1}{n} \sum_{i=1}^{n} m_i$ the estimated average cluster size for the sample.

  $M = \sum_{i=1}^{N} m_i$ the number of elements in the population.

  $\bar{M} = M/N$ the average cluster size for the population.

  y_i the total of all observations in the i-th cluster.

- The estimator of the population mean μ is the sample mean $\bar{y}$, given by

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} m_i}$$

- The estimator has the form of a *ratio estimator*, therefore the estimated variance of $\bar{y}$ is

$$\widehat{Var}(\bar{y}) = \left(\frac{N-n}{Nn\overline{M}^2}\right) \frac{\sum_{i=1}^{n} (y_i - \bar{y}m_i)^2}{n-1}$$

  where the average cluster size for the population ($\overline{M}$) can be estimated by $\bar{m}$ if M (the number of elements in the population) is unknown.

- The estimated variance is *biased*, except if the cluster sizes m_i are equal.

  Anyway, it is a good estimator of $Var(\bar{y})$ if $n \geq 20$.

Example:

A firm is interested in estimating the average per capita income in a certain city. There is not an available list of resident adults. The city is marked off into rectangular blocks, except for two industrial areas and three parks which contain a few houses. The researchers decide that each of the city blocks will be considered a cluster, the two industrial areas will be considered a cluster and, finally, the three parks will be considered a cluster.

The clusters are numbered from 1 to 60 and there is budget for sampling $n = 20$ clusters and to interview every household within each cluster.

Number of	55	60	63	58	71	78	69	58	52	71
Residents m_i	73	64	69	58	63	75	78	51	67	70
Total Income	2210	2390	2430	2380	2760	3110	2780	2370	1990	2810
per cluster y_i	2930	2470	2830	2370	2390	2870	3210	2430	2730	2880

function to cluster sampling programmed in R:

```
cluster.mu <- function(N, m.vec, y, total=T, M=NA) {
# N = number of clusters in population
# M = number of elements in the population
# m.vec = vector of the cluster sizes in the sample
# y = either a vector of totals per cluster, or a list of
#     the observations per cluster (this is set by total)

 n <- length(m.vec)
# If M is unknown, m.bar is estimated with the mean of m.vec
 if(is.na(M)) {mbar <- mean(m.vec)}
 else {mbar <- M/N}

# If there are not totals of observations they are computed
   if(total==F) {y <- unlist(lapply(y,sum))}
```

```
 mu.hat <- sum(y)/sum(m.vec)
 s2.c <- sum((y-(mu.hat*m.vec))^2)/(n-1)
 var.mu.hat <- ((N-n)/(N*n*mbar^2))*s2.c

 B <- 2*sqrt(var.mu.hat)
 cbind(mu.hat,s2.c,var.mu.hat,B)
}

# Example
m <- c(55,60,63,58,71,78,69,58,52,71,
       73,64,69,58,63,75,78,51,67,70)
y <- c(2210,2390,2430,2380,2760,3110,2780,
       2370,1990,2810,2930,2470,2830,2370,
       2390,2870,3210,2430,2730,2880)

cluster.mu(60,m.vec=m,y,total=T,M=NA)
```

- Cluster sampling is an ideal situation to use *pps sampling* (sampling with probabilities proportional to size), since the number of elements in a cluster m_i forms a natural measure of the size of the cluster and it is convenient to sample with probabilities proportional to m_i.

- In this case, $\pi_i = \dfrac{m_i}{M}$ and the estimator of the population mean μ is

$$\hat{\mu}_{pps} = \frac{1}{n} \sum_{i=1}^{n} \bar{y}_i$$

where $\bar{y}_i$ is the mean for the i-th cluster, and the estimated variance of $\hat{\mu}_{pps}$ is

$$\widehat{Var}(\hat{\mu}_{pps}) = \frac{1}{n(n-1)} \sum_{i=1}^{n} (\bar{y}_i - \hat{\mu}_{pps})^2$$

Example: An auditor wishes to estimate the average number of days sick leave per employee over the past quarter. The firm has eight divisions, which varying numbers of employees per division. Since the number of days of sick leave used within each division should be highly correlated with the number of employees, the auditor decides to sample $n = 3$ divisions with probabilities proportional to number of employees.

Division	1	2	3	4	5	6	7	8	**Total**
N of employees	1200	450	2100	860	2840	1910	390	3200	12950

To select a sample with size 3 we can use this command from R:

```
sample(c("1","2","3","4","5","6","7","8"),3, replace=FALSE,
c(1200,450,2100,860,2840,1910,390,3200))
```

Assume that the sample elements are divisions 3, 6 and 8 where the total number of *sick days* are respectively

$$y_i = 4320 \quad y_2 = 4160 \quad y_3 = 5790$$

In this case,

$$\bar{y}_i = \tfrac{4320}{2100} = 2.06 \quad \bar{y}_2 = \tfrac{4160}{1910} = 2.18 \quad \bar{y}_3 = \tfrac{5790}{3200} = 1.81$$

hence,

$$\hat{\mu}_{pps} = \frac{1}{3}\sum_{i=1}^{3} \bar{y}_i = 2.02$$

- The estimated variance of $\hat{\mu}_{pps}$ is

$$\widehat{Var}(\hat{\mu}_{pps}) = \frac{1}{3 \cdot 2}\sum_{i=1}^{3} (\bar{y}_i - \hat{\mu}_{pps})^2 = 0.012$$

- And the interval with a 95% of confidence is

$$2.02 \pm 1.96 \cdot \sqrt{0.012} \Rightarrow [1.8053;\ 2.2347]$$

Cluster Sampling with Stratification

- Cluster sampling can be combined with stratified sampling, because a population can be divided in L strata and a cluster sample can be selected from each stratum.
- As in the case of ratio estimators we can consider *separate estimators* and *combined estimators*.
- Usually the total number of elements in each cluster is not known and we cannot calculate weights. Then, the usual estimators in cluster sampling are the *combined estimators*.

Analysis of a real population

California requires that all students in public schools to be tested each year.

The State Department of Education then puts together the annual Academic Performance Index (*API*), which rates how a school is doing overall in terms of the test scores. Data contains API ratings and demographic information on 6194 schools in 757 school districts.

In this example we will use school districts as the cluster or primary sampling units. We will take a random sample of 189 school districts and look at all of the schools in each one.

Data, in `Stata` format, can be downloaded from

`http://www.ats.ucla.edu/stat/stata/seminars/svy_stata_intro/oscs1`

Program in Stata:

```
* use http://www.ats.ucla.edu/stat/stata/seminars/svy_stata_intro/oscs1, clear


use C:\QM\Eje1Cluster.dta, clear
count


* fpc=757  (total: 757 school districts)
* pw=757/189  (sample of 189 districts)
* dnum: Identification number of each district


svyset dnum [pweight=pw], fpc(fpc)
svydes


svy: mean api00
svy: total stype
```

```
* Compute the average proportion of English language learners
* and students eligible for subsidized school meals for elementary,
* middle, and high schools


svy: mean ell meals, over(stype)


* Regression models show that these socioeconomic variables
* predict API score and whether the school achieved
* its API target


svy: reg api00 ell meals
```

Program in R:

```
# Import data from Stata format

library(foreign)

thing <- read.dta("C:/QM/Eje1Cluster.dta")


library(survey)

dclus1 <- svydesign(ids=~dnum, weight=~pw, data=thing, fpc=~fpc)

summary(dclus1)


svymean(~api00, dclus1)

svyquantile(~api00, dclus1, quantile=c(0.25,0.5,0.75), ci=TRUE)


svytotal(~stype, dclus1)
```

```r
# Compute the average proportion of English language learners
# and students eligible for subsidized school meals for elementary,
# middle, and high schools

svyby(~ell+meals, ~stype, design=dclus1, svymean)

# Regression models show that these socioeconomic variables
# predict API score and whether the school achieved
# its API target

regmodel <- svyglm(api00 ~ ell + meals, design=dclus1)
summary(regmodel)
```

Observations:

- With *cluster sampling*, the smaller the size of the clusters the better is. When there is a hierarchy of clusters, the smallest ones will generally be the preferred choice.
- For example, in a High School example, the students could be grouped by grade levels or classes; here grade levels are too large to serve as clusters for sampling purposes, and classes are the obvious choice.
- The problem with cluster sampling is that, because clusters usually comprise existing groupings that were formed for other purposes, the lowest level of clustering still often yields clusters that are too large to be used efficiently in cluster sampling.
- The solution to this problem is to divide the clusters into sub-clusters for sampling purposes; essentially this is what is done in *multistage sampling*.

Two-Stage Cluster Sampling

- A two-stage cluster sample is obtained by first selecting a sample of clusters, and then selecting a sample of elements from each sampled cluster.
- There are two desirable conditions for selecting appropriate clusters:
  - Geographic proximity of the elements within a cluster
  - Cluster sizes that are convenient to handle.
- Example: Consider an university student opinion poll.
  - If the students in an university hold similar opinions but differ widely from university to university, the sample should contain few representatives from many different universities.
  - If the students in an university vary greatly, the sample should contain many representatives from each of a few universities.

- In two-stage cluster sampling, the sample of elements is obtained as a result of two stages of sampling.
- The population elements are first grouped into disjoint subpopulations, called *primary sampling units* (**PSU**). Then, in a first-stage sampling, a sample of PSU is drawn.
- In the second-stage sampling units (**SSU**) may be clusters of elements, for each PSU in the first-stage sample.
- A sample of SSU is drawn (second-stage sampling) from each PSU in the first-stage sample. When the SSU are clusters, every element in the selected SSU is surveyed.

Example of cluster sampling

The *Swedish Board of Education* take annual surveys in Sweden to measure drug use among youngster students. Data on drug use is collected through anonymous questionnaires from every student in a sample of *ninth-grade* classes. The sampling frame consists of a list of **all** ninth-grade classes.

Example of two-stage cluster sampling, with schools as **PSU** and with classes as **SSU**:

(*i*) A sample of schools is drawn from a frame containing all the schools in the country.

(*ii*) From every selected school, a sample of ninth-grade classes is drawn and all students in the selected classes are surveyed.

We use the **following notation**:

- N the number of clusters in the population.
- n the number of clusters *selected* in a simple random sample.
- M_i the number of elements in cluster i.
- m_i the number of elements selected in a simple random sample from cluster i.
- $M = \sum_{i=1}^{N} M_i$ the number of elements in the population.
- $\bar{M} = M/N$ the average cluster size for the population.
- y_{ij} the j-th observation in the sample from the i-th cluster.
- $\bar{y}_i = \frac{1}{m_i} \sum_{j=1}^{m_i} y_{ij}$ the sample mean for the i-th cluster.

The estimator of the **population mean** μ is

$$\hat{\mu} = \left(\frac{N}{M}\right) \frac{\sum_{i=1}^{n} M_i \bar{y}_i}{n}$$

assuming simple random sampling in each stage.

The **estimated variance** of $\hat{\mu}$ is

$$\widehat{Var}(\hat{\mu}) = \left(\frac{N-n}{N}\right) \frac{1}{n\bar{M}^2} s_b^2 + \frac{1}{nN\bar{M}^2} \sum_{i=1}^{n} M_i^2 \left(\frac{M_i - m_i}{M_i}\right) \frac{s_i^2}{m_i}$$

where

$$\begin{aligned} s_b^2 &= \frac{\sum_{i=1}^{n} \left(M_i \bar{y}_i - \bar{M}\hat{\mu}\right)^2}{n-1} \\ s_i^2 &= \frac{\sum_{j=1}^{m_i} \left(y_{ij} - \bar{y}_i\right)^2}{m_i - 1}, \qquad i = 1, 2, \ldots, n \end{aligned}$$

When the total number of elements in the population M is **not known** it is estimated by

$$\hat{M} = N \cdot \frac{\sum_{i=1}^{n} M_i}{n}$$

that is, the average cluster size multiplied by the number of clusters in the population.

In this case, the estimator of the population mean μ is

$$\hat{\mu} = \frac{\sum_{i=1}^{n} M_i \bar{y}_i}{\sum_{i=1}^{n} M_i}$$

The estimated variance of $\hat{\mu}$ is the same as before, but substituting s_b^2 by s_r^2:

$$s_r^2 = \frac{\sum_{i=1}^{n} M_i^2 \left(\bar{y}_i - \hat{\mu}\right)^2}{n-1}$$

Example:

The entertainment spent is important to the businesses of a town with a residence for students. A firm is interested in estimating the average monthly amount spent on entertainment per student. The sampling procedure that they use is to locate randomly selected rooms and to sample a subset of students in those rooms. The number of rooms is 112 with a total number of 306 students. Data are shown in the table:

M_i	3	2	4	3	6	3	5	4	6	3
m_i	2	2	2	2	2	2	2	2	3	2
y_i	23, 27	18, 3	37, 12	22, 21	18, 62	21, 25	3, 12	7, 36	2, 12, 51	18, 17

See a **function** to make a two-stage cluster sampling programmed in `R`:

```
two.stage.cluster <- function(N,n,y,M=NA,Mi,mi) {
 # N = number of clusters in population
 # n = number of clusters sampled
 # y = a list of the observations from sampled clusters
 # M = number of elements in population, often unknown
 # Mi = number of elements in each of the selected clusters
 # mi = number of elements sampled in each selected cluster


   yi.bar <- unlist(lapply(y,mean))
   s2i    <- unlist(lapply(y,var))
     if(is.na(M)) {    # case where M is unknown
      M.bar <- mean(Mi)
      mu.hat <- sum(Mi*yi.bar)/(n*M.bar)
      s2.r <- sum((Mi^2)*(yi.bar-mu.hat)^2)/(n-1)
      thing.1 <- ((N-n)/N)*(1/(n*M.bar^2))*s2.r
      thing.2 <- (1/(n*N*(M.bar^2)))*sum(Mi^2*((Mi-mi)/Mi)*(s2i/mi))
      var.mu <- thing.1 + thing.2
```

```
    B <- 2*sqrt(var.mu)
    cat(" mu.hat =",mu.hat,"\n", " var.mu =",var.mu,"\n",
     " B =",B,"\n", " yi.bar =",yi.bar,"\n", " s2i =",s2i,"\n",
     " s2.r =",s2.r,"\n")
  } else {   # case where M is known
    M.bar <- M/N
    mu.hat <- (1/M.bar)*(sum(Mi*yi.bar)/n)
    s2.b <- sum((Mi*yi.bar-M.bar*mu.hat)^2)/(n-1)
    thing.1 <- ((N-n)/N)*(1/(n*M.bar^2))*s2.b
    thing.2 <- (1/(N*n*(M.bar^2)))*sum(Mi^2*((Mi-mi)/Mi)*(s2i/mi))
    var.mu <- thing.1 + thing.2
    B <- 2*sqrt(var.mu)
     cat("  mu.hat =",mu.hat,"\n", " var.mu =",var.mu,"\n"," B =",B,"\n",
     " yi.bar =",yi.bar,"\n", " s2i =",s2i,"\n"," s2.b =",s2.b,"\n")
    }
}
```

```
N <- 112

n <- 10

M <- 306


Mi <- c(3,2,4,3,6,3,5,4,6,3)

mi <- c(2,2,2,2,2,2,2,2,3,2)


y <- list(y1=c(23,27),y2=c(18,3),y3=c(37,12),y4=c(22,21),y5=c(18,62),

   y6=c(21,25),y7=c(3,12),y8=c(7,36),y9=c(2,12,51),y10=c(18,17))


two.stage.cluster(N,n,y,M,Mi,mi)
```

STEPS FOR PERFORMING A DESIGN-BASED ANALYSIS

1. Identify the following elements of the sample design:
    - Stratification.
    - Clustering variables used.
    - Population sizes required for determination of finite population corrections.
2. On the basis of above information, determine the sampling weight for each sample subject.
3. Determine for each sample unit a final sampling weight that takes into consideration any non-response and post-stratification adjustments that are desired.
4. Ensure that all stratification, clustering and population size data required for an appropriate design-based analysis are identified on each sample unit.
5. Interpret results and findings.

Analysis of a real population

- We take a stratified two-stage cluster sample. The sampling for each stratum is done independently with respect to other strata. Clusters are sampled, and then elements within each of the selected clusters are also sampled.
- We take a *SRS* of school districts (*clusters*), and then we take a *SRS* of schools (*elements*).
- We first stratify schools based on their mean `api99score`. Next, we randomly select 25% of the school districts from each strata. Finally, we randomly select three schools from each selected district.

  Data, in `Stata` format, can be downloaded from

  `http://www.ats.ucla.edu/stat/stata/seminars/svy_stata_intro/strataboth`

Program in Stata:

```
* use http://www.ats.ucla.edu/stat/stata/seminars/svy_stata_intro/strataboth, clear


use C:\QM\Eje2Cluster.dta, clear
count


* dnum: Identification number of each district


svyset dnum [pweight=pwt], fpc(fpc) strata(strata)
svydes


svy: mean api00 growth
svy: total yr_rnd
```

```
* Compute the average proportion of English language learners and
* of students eligible for subsidized school meals for each stratum


svy: mean ell meals, over(strata)


* Regression models show that these socioeconomic variables
* predict API score and whether the school achieved
* its API target


svy: reg api00 awards meals
```

Program in R:

```
# Import data from Stata format

library(foreign)

thing <- read.dta("C:/QM/Eje2Cluster.dta")


library(survey)


# Stratified 2-level Cluster Sampling design

dclus2 <- svydesign(ids=~dnum+~snum, strata=~strata, data=thing, weight=~pwt, fpc=~fpc+N)

summary(dclus2)


eso <- svymean(~api00, dclus2)

supe <- eso[[1]] + qt(0.975,187)*sqrt(attr(eso,"var"))

infe <- eso[[1]] - qt(0.975,187)*sqrt(attr(eso,"var"))
```

```r
print(eso)

cat("Confidence Interval for api00: [",infe,";",supe,"]","\n")

# Normal approximation

confint(eso)



eso <- svymean(~growth, dclus2)

supe <- eso[[1]] + qt(0.975,187)*sqrt(attr(eso,"var"))

infe <- eso[[1]] - qt(0.975,187)*sqrt(attr(eso,"var"))

print(eso)

cat("Confidence Interval for growth: [",infe,";",supe,"]","\n")

# Normal approximation

confint(eso)
```

```r
eso <- svytotal(~yr_rnd, dclus2)

supe <- eso[[1]] + qt(0.975,187)*sqrt(attr(eso,"var"))

infe <- eso[[1]]- qt(0.975,187)*sqrt(attr(eso,"var"))

print(eso)

cat("Confidence Interval for yr_rnd: [",infe,";",supe,"]","\n")

# Normal approximation

confint(eso)


# Compute the average proportion of English language learners and

# students eligible for subsidized school meals for elementary,

# middle, and high schools


svyby(~ell+meals, ~strata, design=dclus2, svymean, vartype="ci")
```

```r
# Regression models show that these socioeconomic variables

# predict API score and whether the school achieved

# its API target


regmodel <- svyglm(api00 ~ awards + meals, design=dclus2)

summary(regmodel)


X11()

svyboxplot(~api00 ~ factor(strata), dclus2, col="peachpuff")
```