Sampling and Survey Weights

Survey Research Design and Analysis

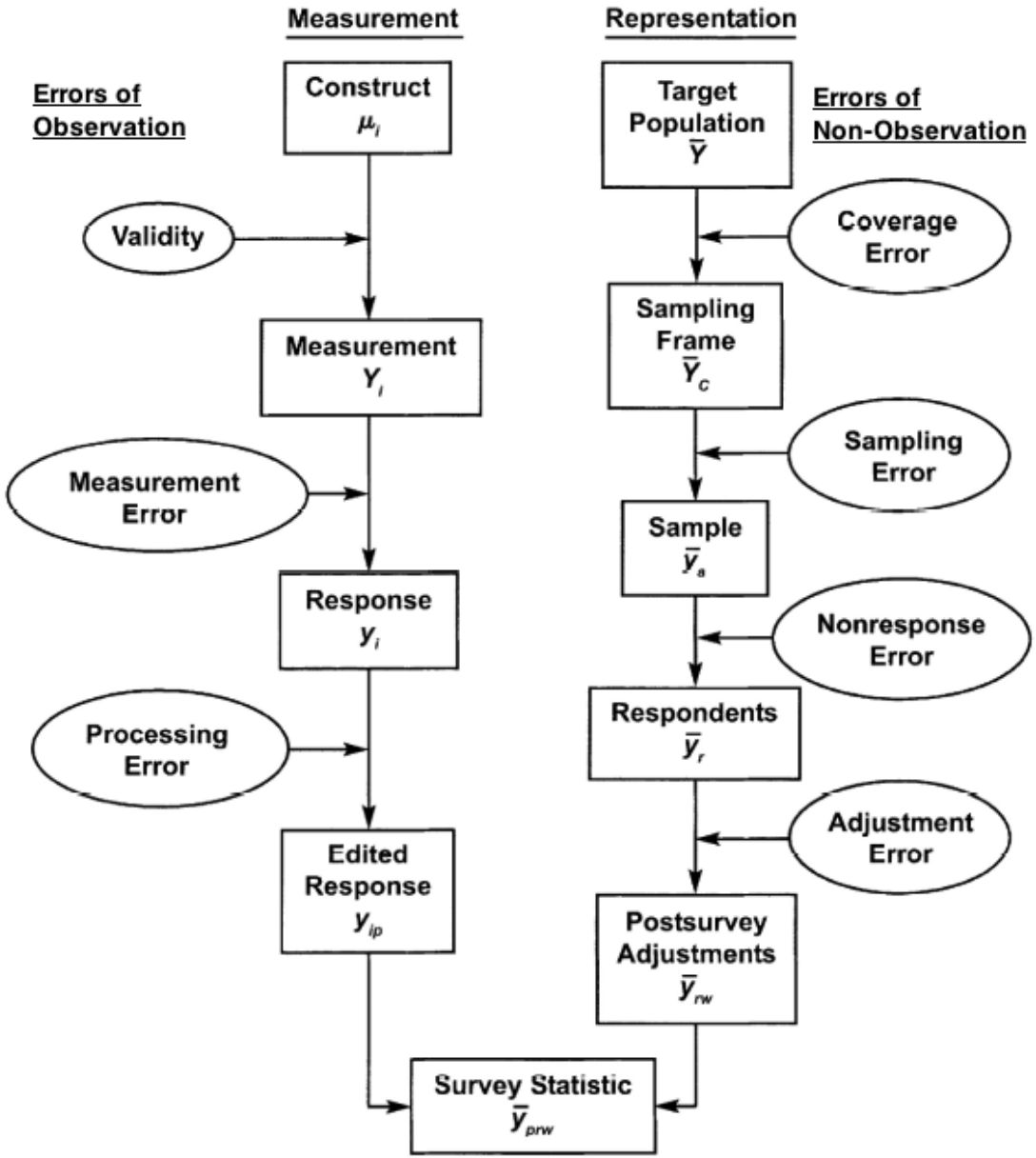
Soledad Artiz Prillaman
Oxford University

May 20, 2019

Outline

Sampling

Now that we have figured out how to define our sample frame, and reduced coverage error, we have to actually draw our sample!


Measurement
Representation
Errors of Observation
Construct μ_i
Target Population $\bar{Y}$
Errors of Non-Observation
Validity
Coverage Error
Measurement Y_i
Sampling Frame $\bar{Y}_C$
Measurement Error
Sampling Error
Sample $\bar{y}_a$
Response y_i
Nonresponse Error
Respondents $\bar{y}_r$
Processing Error
Adjustment Error
Edited Response y_{ip}
Postsurvey Adjustments $\bar{y}_{rw}$
Survey Statistic $\bar{y}_{prw}$

Sampling Error

Sampling Error: when the sample does not fully and accurately represent sampling frame.

Can affect:

- Sampling bias
- Sampling variance

Sampling Error

Function of:

- Probability sampling
- Stratification
- Clustering
- Sample size

Sampling Bias

Sampling Bias: when the average mean of *repeated* samples does not equal mean of the sampling frame.Arises when some members of the sampling frame are given no chance (reduced chance) of selection.

$$\frac{1}{C}\sum_{i=1}^{C} Y_i \neq \frac{1}{n}\sum_{i=1}^{n} y_i$$

- C - number covered units in target population
- n - size of sample

Sampling Bias

Sampling Bias: when the average mean of *repeated* samples does not equal mean of the sampling frame. Arises when some members of the sampling frame are given no chance (reduced chance) of selection.

$$\frac{1}{C}\sum_{i=1}^{C} Y_i \neq \frac{1}{n}\sum_{i=1}^{n} y_i$$

Mainly affected by how probabilities of selection are assigned. Solved with equal probabilities (probability sampling).
$\Rightarrow$ Function of:

- Probability sampling

Sampling Variance

Sampling Variance: the variance of the sampling distribution.

$$Var(\bar{X}) = \frac{1}{n}S^2$$

- S - sample standard deviation
- n - size of sample

Sampling Variance

Sampling Variance: the variance of the sampling distribution.

$$Var(\bar{X}) = \frac{1}{n}S^2$$

Reduced as S decreases or n increases.
$\Rightarrow$ Function of:

- Stratification
- Clustering
- Sample size

Non-Probability Sampling

- Purposeful sampling
- Snowball sampling
- Convenience sampling
- Quota sampling

Example: Street intercept surveys.

No theoretical reason to believe these generate unbiased samples!

Probability Sampling

Probability Sampling: when all sampling frame elements have known, nonzero chances of selections into the sample.

Probabilities do not need to be equal. Just known.

No bias **but does not guarantee that any given sample will be representative.**

Survey Weights

Survey weights allow for our estimates to reflect the actual distribution/characteristics of the population.

Why Weight?

To improve the representativeness of the sample!

When Weight?

- Different probabilities of selection
- High nonresponse or coverage errors
- Non-representative sample
- Small sample size

Types of Survey Weights

- Nonresponse Weights
- Design Weights
- Poststratification Weights

Design Weights

Probability Sampling: when each element of the sample frame has a known and nonzero probability of selection into the sample.

Inclusion Probability: probability of being included in the sample. Can be different for each element of the sample frame.

If equal probabilities for every element in the sample frame (Simple Random Sample):

$$\pi_i = \frac{n}{N}$$

- π_i - Inclusion Probability for individual i
- n - sample size
- N - population size

Design Weights

Design Weight: How many units in the population are represented by each element in the sample. I.e. how many people in the population are represented by each sample person.

$$W_i = \frac{1}{\pi_i}$$

If equal probabilities for every element in the sample frame (Simple Random Sample):

$$W_i = \frac{N}{n}$$

- W_i - Design Weight for individual i

Design Weights

Design Weight: How many units in the population are represented by each element in the sample. I.e. how many people in the population are represented by each sample person.

$$W_i = \frac{1}{\pi_i}$$

The sum of sampling weights is equal to the population size N. Why?

Horvitz-Thomson Estimator

$$\hat{Y} = \sum_{i=1}^{n} W_i y_i$$

Example

$$\begin{aligned} N &= 300,000 \\ n &= 300 \\ \sum_{i=1}^{n} y_i &= 150 \end{aligned}$$

What is our estimate for Y assuming equal probability of selection for all i?

$$\hat{Y} = \frac{300,000}{300} \times 150 = 150,000$$

Design Weights in R: Survey Package

Steps:

1. Calculate/identify your design weights
2. Specify your survey design with `svydesign()`
3. Estimate your statistic using a survey model such as `svymean()` or `svyglm()`

Design Weights in R: Survey Package

```
library(survey)
data(api)

# 1. Calculate/identify your design weights
summary(apisrs$pw)

# 2. Specify your survey design
srs_design <- svydesign(id=~1, weights=~pw, data=apisrs)


# 3. Estimate your statistic using a survey model
svymean(~enroll, srs_design)
svyglm(enroll ~ api00 + stype,
       design = srs_design)
svyglm(awards_num ~ api00 + stype,
       design = srs_design,
       family = binomial(link=logit))
```

Design Weights in R: Zelig Package

Steps:

1. Calculate/identify your design weights
2. Estimate your statistic using a survey model with `zelig()`

Design Weights in R: Zelig Package

```
library(survey)
data(api)

# 1. Calculate/identify your design weights
summary(apisrs$pw)

# 2. Estimate your statistic using a survey model
zelig(enroll ~ api00 + stype,
        weights = apisrs$pw,
        model = "normal.survey",
        data = apisrs)
zelig(awards_num ~ api00 + stype,
        weights = apisrs$pw,
        model = "logit.survey",
        data = apisrs)
```

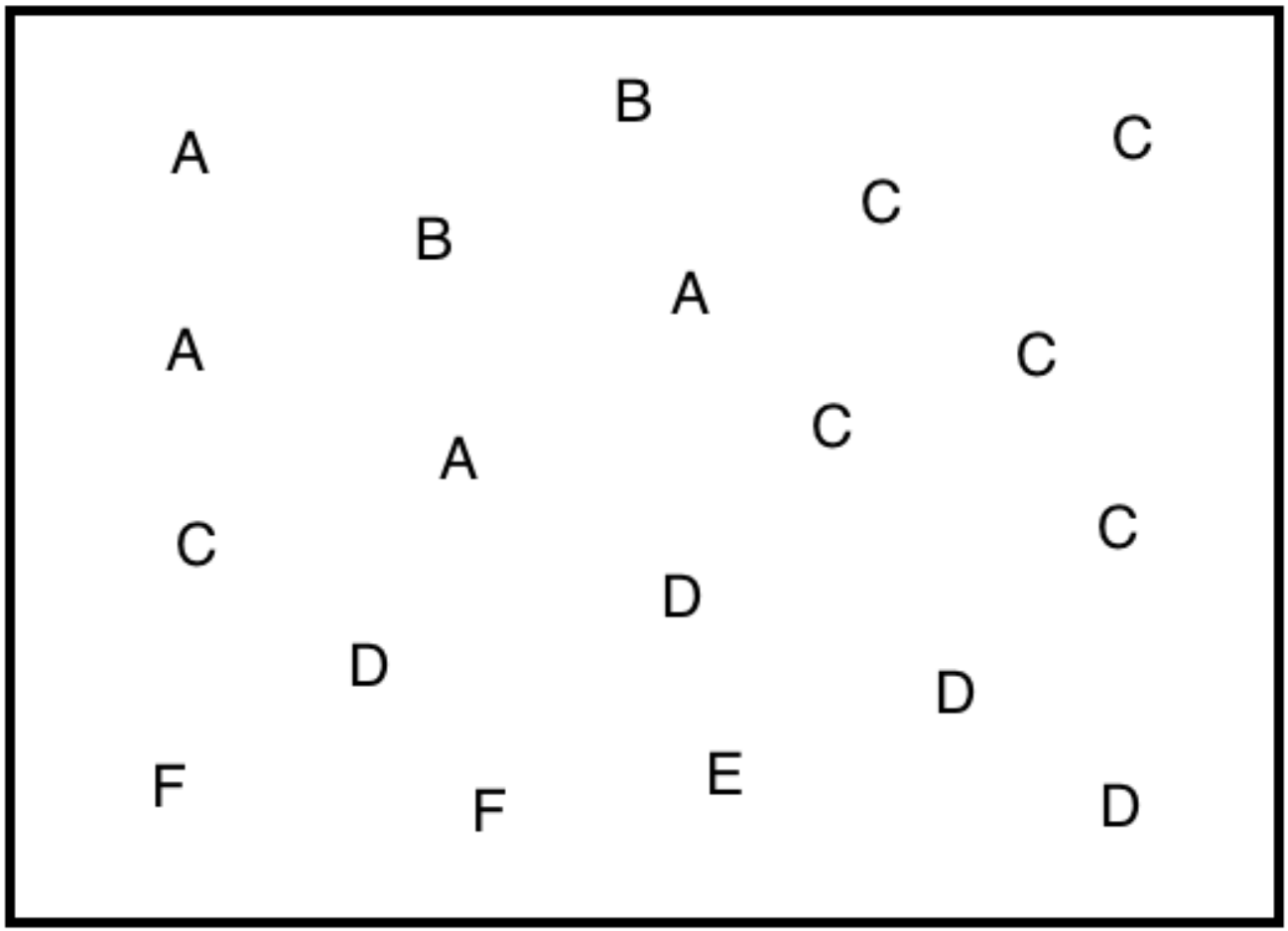
Outline

Simple Random Sampling

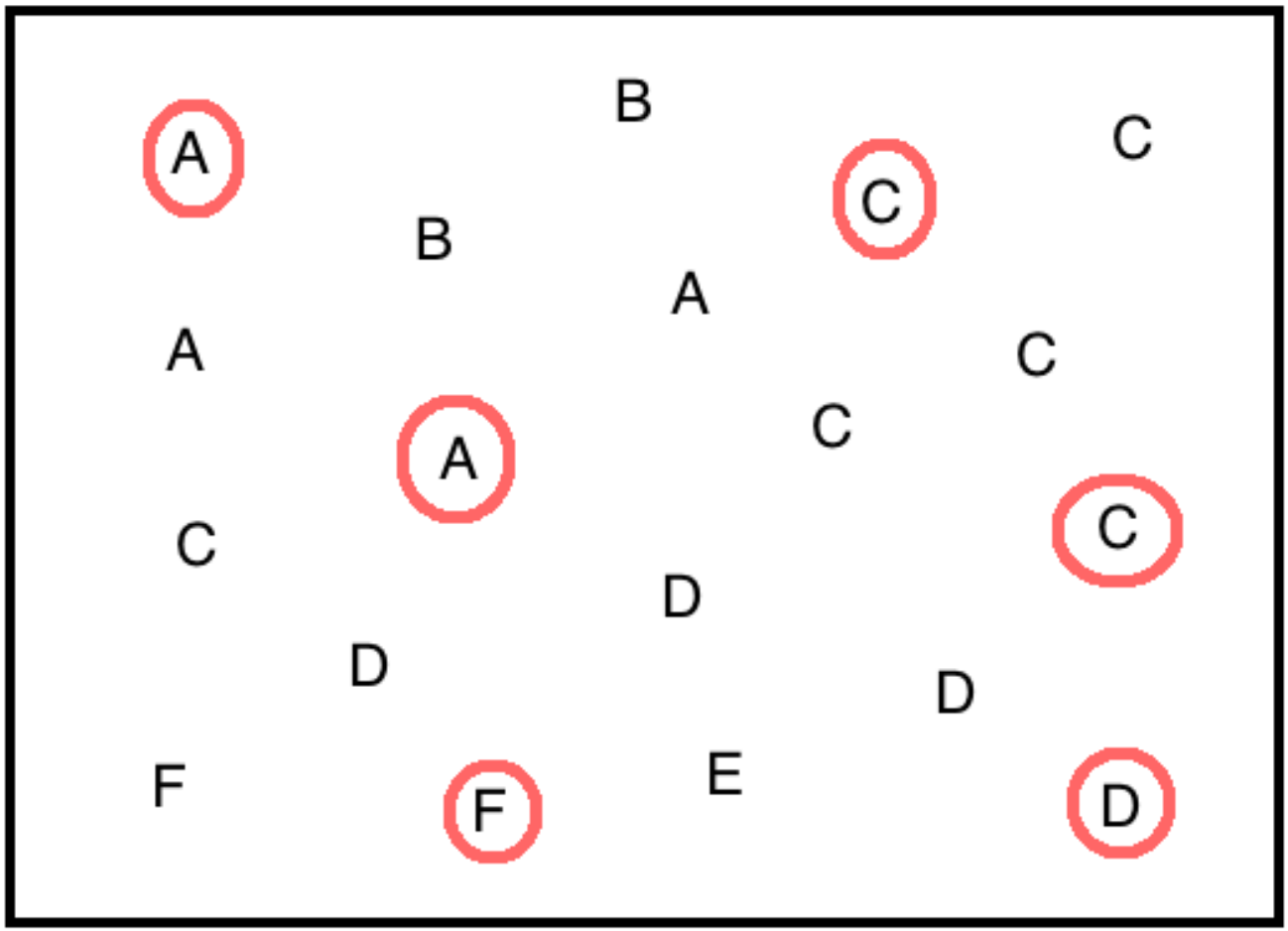
Simple Random Sampling (SRS): assign an equal probability of selection to each element in the sampling frame and equal probability to all pairs of elements.

- Select random numbers to apply to all elements of the sample frame (list)
- Sample without replacement

Simple Random Sampling



Simple Random Sampling



Simple Random Sampling

Sample mean:

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

Variance of the sample mean:

$$v(\bar{y}) = \frac{(1-f)}{n} s^2$$

- $(1-f)$ - finite population correction (fpc). Proportion of frame elements not sampled.

  $f = \frac{n}{N}$

  Disappears for N large.
- s^2 - sample standard deviation (standard deviation of observed y_is)

Simple Random Sampling

Advantages?

- Simple to sample
- Simple to analyze

Disadvantages?

- May not be representative
- May be able to reduce sampling variance
- Costly

Simple Random Sampling in R

```
# First, let's draw our sample
set.seed(1234)
my_srs <- apipop[srswor(n = 1000, N = pop_N)==1,]

# Let's tell R that this our srs is survey data
my_srs$pop_N <- rep(pop_N,dim(my_srs)[1])
srs_design <- svydesign(id = ~1, fpc = ~pop_N, data=my_srs)

# Let's estimate the population mean for the variable api00
srs_mean <- svymean(~api00, srs_design)
# This gives us the mean of 663.85 and standard error of 3.704

# How were these numbers calculated?
# Remember that mu_hat = 1/n sum x
n <- length(my_srs$api00)
mu_hat = 1/n * sum(my_srs$api00)
# And V(mu_hat) = (N-n)/n V(x)
v_mu_hat = ((pop_N-n)/pop_N) * (var(my_srs$api00)/n)
sqrt(v_mu_hat)
# It's the same!
```

Weights: Simple Random Sampling

Since all elements in the sample frame have equal probabilities of selection:

$$\begin{aligned} \pi_i &= \frac{n}{N} \\ W_i &= \frac{N}{n} \end{aligned}$$

Calculating SRS Weights in R

```
library(survey)
data(api)

# The pw variable store the design weight for each row
summary(apisrs$pw)

# These weights are the same for every observation. Why?

# Can we figure out where they came from?
N <- dim(apipop)[1] # Store the total population size
n <- dim(apisrs)[1] # Store the SRS sample size
N/n
# Voila!
```

Outline

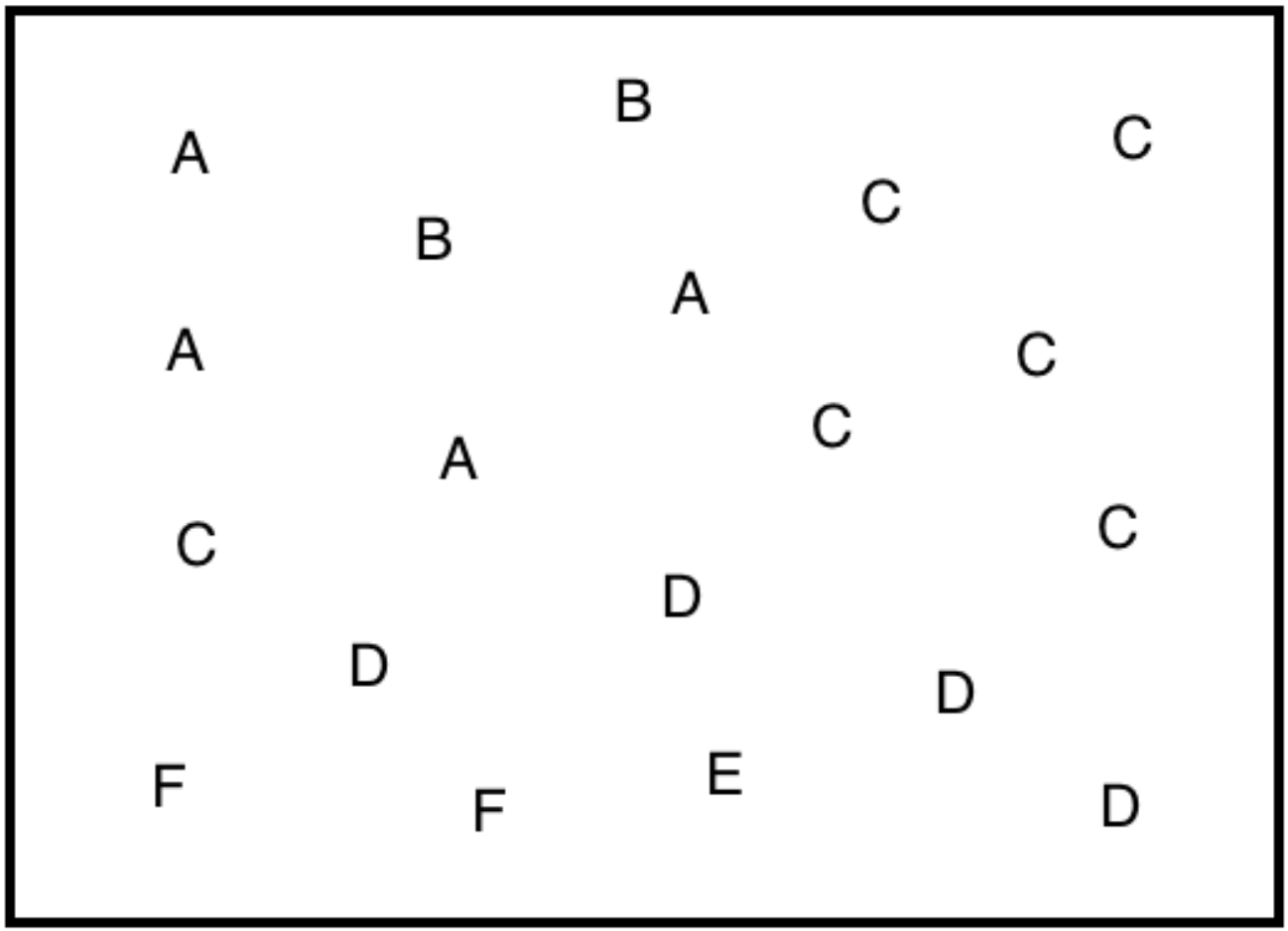
Stratified Random Sampling

Stratified Random Sampling: assign an equal probability of selection to each element *within each strata* in the sampling frame. Sample units from each strata (via SRS or possibly clustering as we will later see).

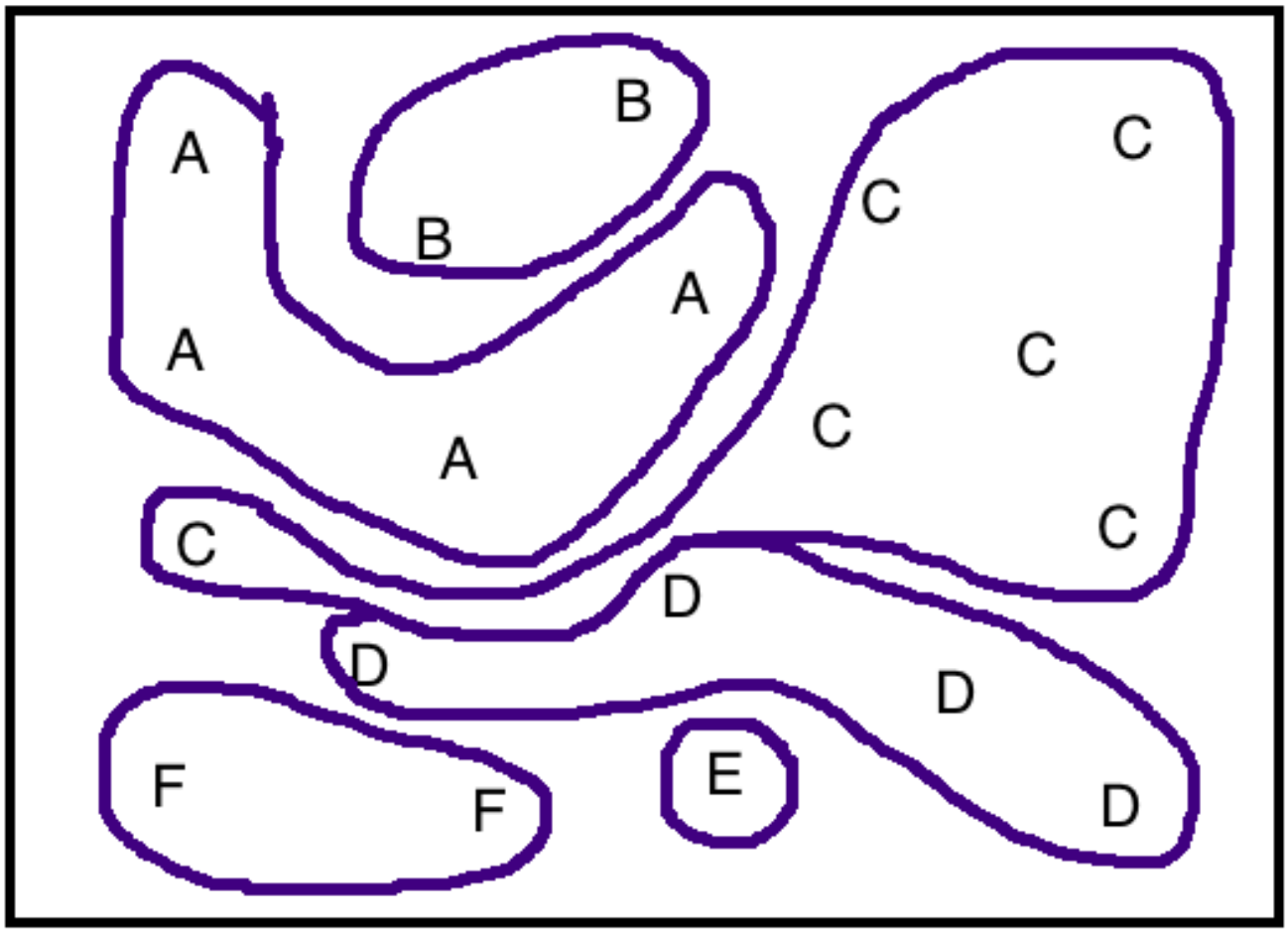
Strata: mutually exclusive groups of elements on a sampling frame that divide the sampling frame.

- Group elements of sampling frame into strata
- Within each strata, sample without replacement (via SRS)

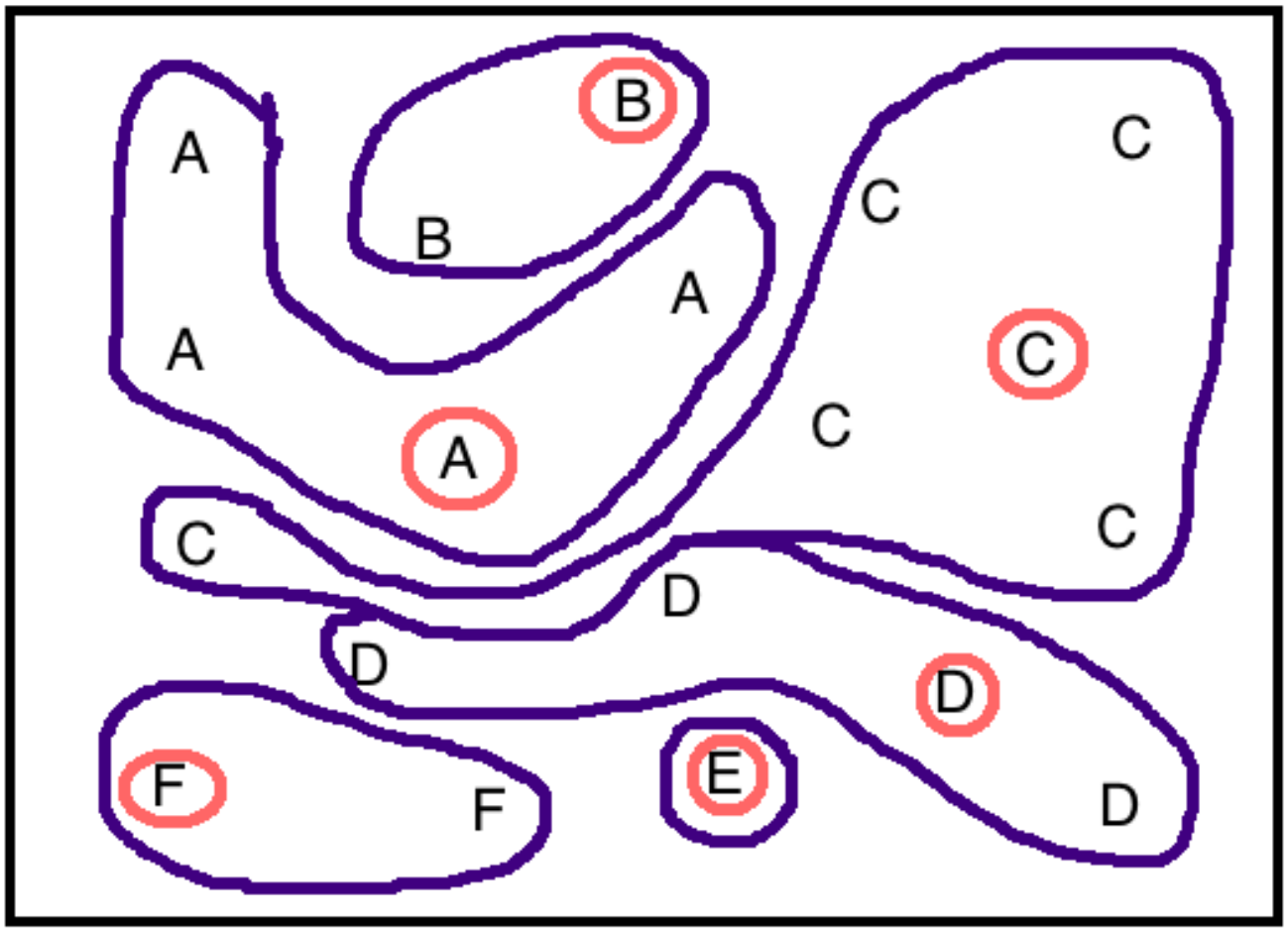
Stratified Random Sampling



Stratified Random Sampling



Stratified Random Sampling



How do we allocate the sample to strata?

We want to take a sample of n. How do we allocate this across the strata? I.e. how do we decide on the sample size within each strata, or n_h?

Proportionate Allocation to Strata: select the sample in each stratum with the same probability of selection in the population. The proportion of sampled elements from a given stratum is the same as the proportion of elements in the population in that stratum.

$$\frac{n_h}{n} = \frac{N_h}{N}$$

- n_h - sample size within stratum h
- n - total sample size across all strata
- N_h - number of population elements in stratum h
- N - total population size

How do we allocate the sample to strata?

We want to take a sample of n. How do we allocate this across the strata? I.e. how do we decide on the sample size within each strata, or n_h?

Optimal Allocation to Strata: select the sample in each stratum in relation to the variability of of units within stratum. Oversample with a statum if the stratum is large or has high within stratum variance.

$$\frac{n_h}{n} = \frac{\frac{N_h S_h}{\sqrt{c_h}}}{\sum_{l=1}^{H} \frac{N_l S_l}{\sqrt{c_l}}}$$

- N_h - number of population elements in stratum h
- S_h - variance within stratum h
- c_h - cost to sample a unit from stratum h

Stratified Random Sampling

How can we calculate the sample mean?
Weight the stratum results!

$$\bar{y} = \sum_{h=1}^{H} W_h \bar{y}_h$$

$$\bar{y}_h = \frac{1}{n_h} \sum_{i=1}^{n_h} y_i$$

What is W_h? Weight given to each stratum. Under proportionate allocation:

$$W_h = \frac{N_h}{N}$$

Stratified Random Sampling

How can we calculate the variance of the sample mean?
Weight the stratum results!

$$v(\bar{y}) = \sum_{h=1}^{H} W_h^2 v(\bar{y}_h)$$

$$v(\bar{y}_h) = \frac{(1 - f_h)}{n_h} s_h^2$$

How does this compare to SRS?

Design Effect (d^2): the ratio of the sampling variance for a statistic computed under the sample design divided by the sampling variance that would have been obtained from an SRS sample. An indicator of whether the sampling variance from our sample is bigger or smaller that the sampling variance under SRS.

$$d^2 = \frac{v(\bar{y})}{v_{srs}(\bar{y})}$$

How does this compare to SRS?

We can estimate the design effect for our stratified random sample with:

$$d^2 = \frac{\sum_{h=1}^{H} W_h^2 \left(\frac{1-f_h}{n_h}\right) s_h^2}{\left(\frac{1-f}{n}\right) s^2}$$

- $d^2 < 1 \rightarrow$ variance under stratified sampling is **lower**
- $d^2 > 1 \rightarrow$ variance under stratified sampling is **higher**
- $d^2 = 1 \rightarrow$ variance under stratified sampling is **the same**

Why and When Stratify?

Why stratify?

- Gains to precision!

When subpopulations (strata) are:

- identifiable
- differ from one another; ie. groups are meaningful
- correlate with variable of interest
- have low within-stratum variance (Why?)

Stratified Random Sampling

Advantages?

- Often lower variance
- Guaranteed representativeness of subpopulations

Disadvantages?

- Need complete sampling frame
- Potentially costly
- Slightly more complicated

STRATIFIED RANDOM SAMPLING IN R

```
# We will use stype as the  strata variable - School Type

# 1. Figure out the sizes of the population for each strata
pop_strata_N <- table(apipop$stype)

# 2. Calculate strata sample sizes under proportionate
    allocation
strata_N <- round(1000*(pop_strata_N/pop_N))

# 3. Sample
set.seed(1234)
strata_sample <- strata(apipop,stratanames=c("stype"),
    size=strata_N, method="srswor")
my_strat <- getdata(apipop, strata_sample)

# 4. Tell R that this is survey data
strat_design <- svydesign(id=~1, fpc=~pop_strata_N,
    strata=~stype, data=my_strat)

# 5. estimate the population mean for the variable enroll
strat_mean <- svymean(~api00, strat_design)
# Mean of 639.67 and standard error of 3.2678
```

Weights: Stratified Random Sampling

Proportionate Allocation to Strata: select the sample in each stratum with the same probability of selection in the population. The proportion of sampled elements from a given stratum is the same as the proportion of elements in the population in that stratum.

$$\frac{n_h}{n} = \frac{N_h}{N}$$

- n_h - sample size within stratum h
- n - total sample size across all strata
- N_h - number of population elements in stratum h
- N - total population size

Weights: Stratified Random Sampling

Since all elements in each strata in the sample frame have equal probabilities of selection:

$$\pi_{i,h} = \frac{n_h}{N_h}$$

$$W_{i,h} = \frac{N_h}{n_h}$$

Calculating Stratified Weights in R

```
library(survey)
data(api)

# The pw variable store the design weight for each row
table(apistrat$pw, apistrat$stype)

# These weights are the same for every observation
within a strata. Why?

# Can we figure out where they came from?
# Store population size in each strata
N <- table(apipop$stype)
# Store the sample size in each strata
n <- table(apistrat$stype)
N/n
# Voila!
```

Outline

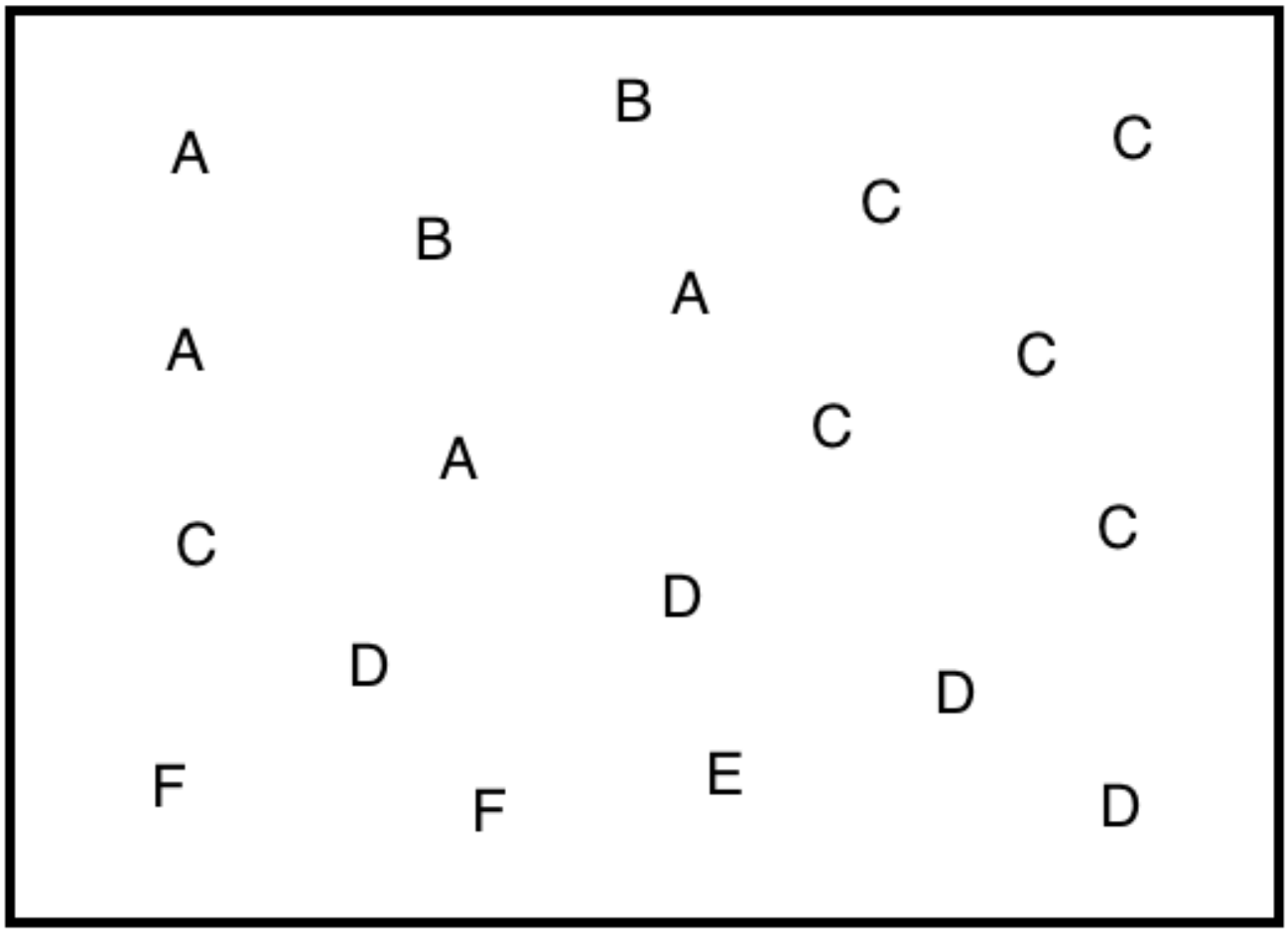
Clustered Random Sampling

Clustered Random Sampling: assign an equal probability of selection to each *group of elements* in the sampling frame.
Clusters: mutually exclusive groups of elements on a sampling frame that divide the sampling frame.

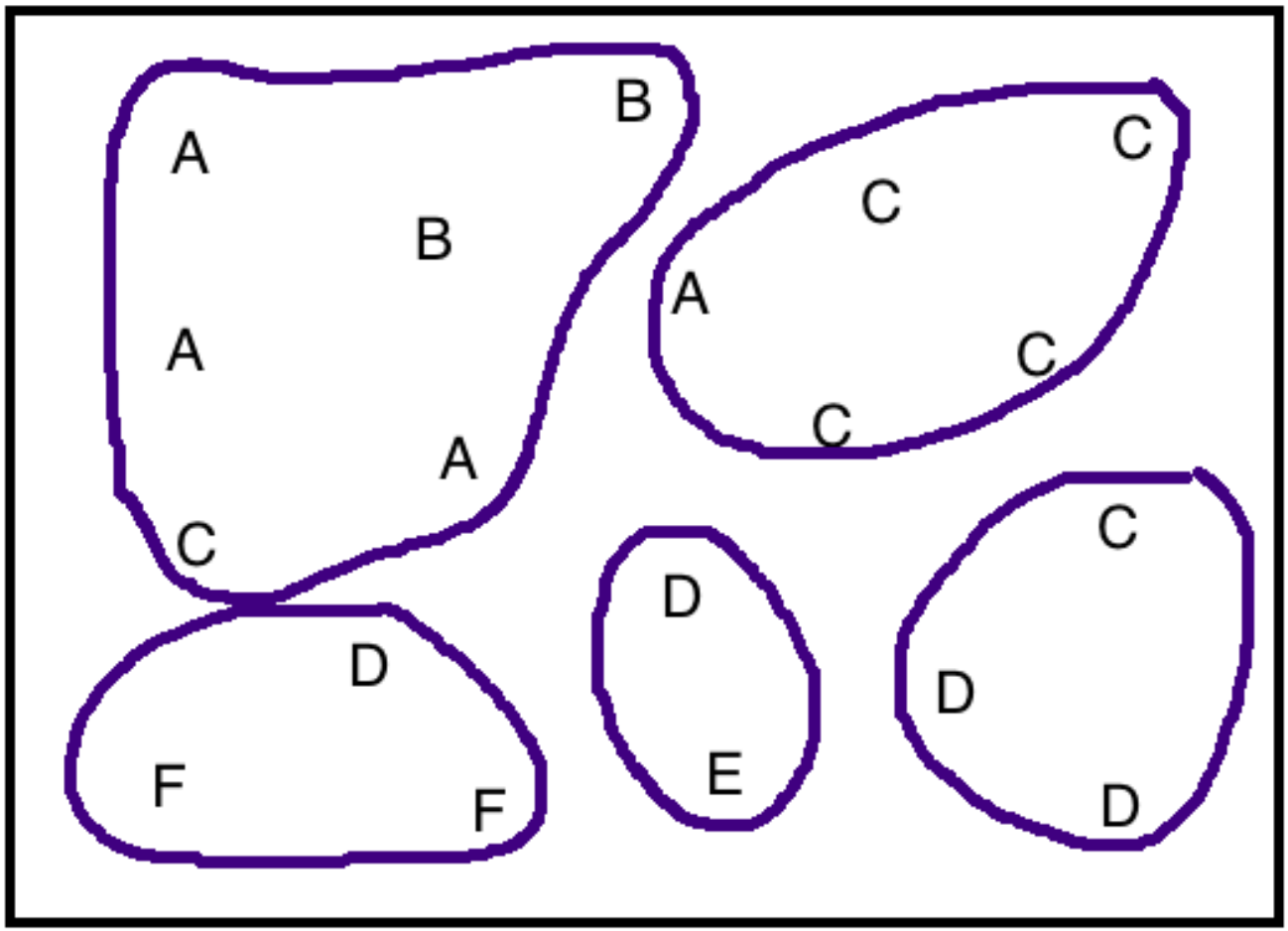
- Group elements of sampling frame into clusters
- Randomly sample clusters using SRS
- Sample all elements from within the sampled clusters

Clusters are the Primary Sampling Unit (PSU).

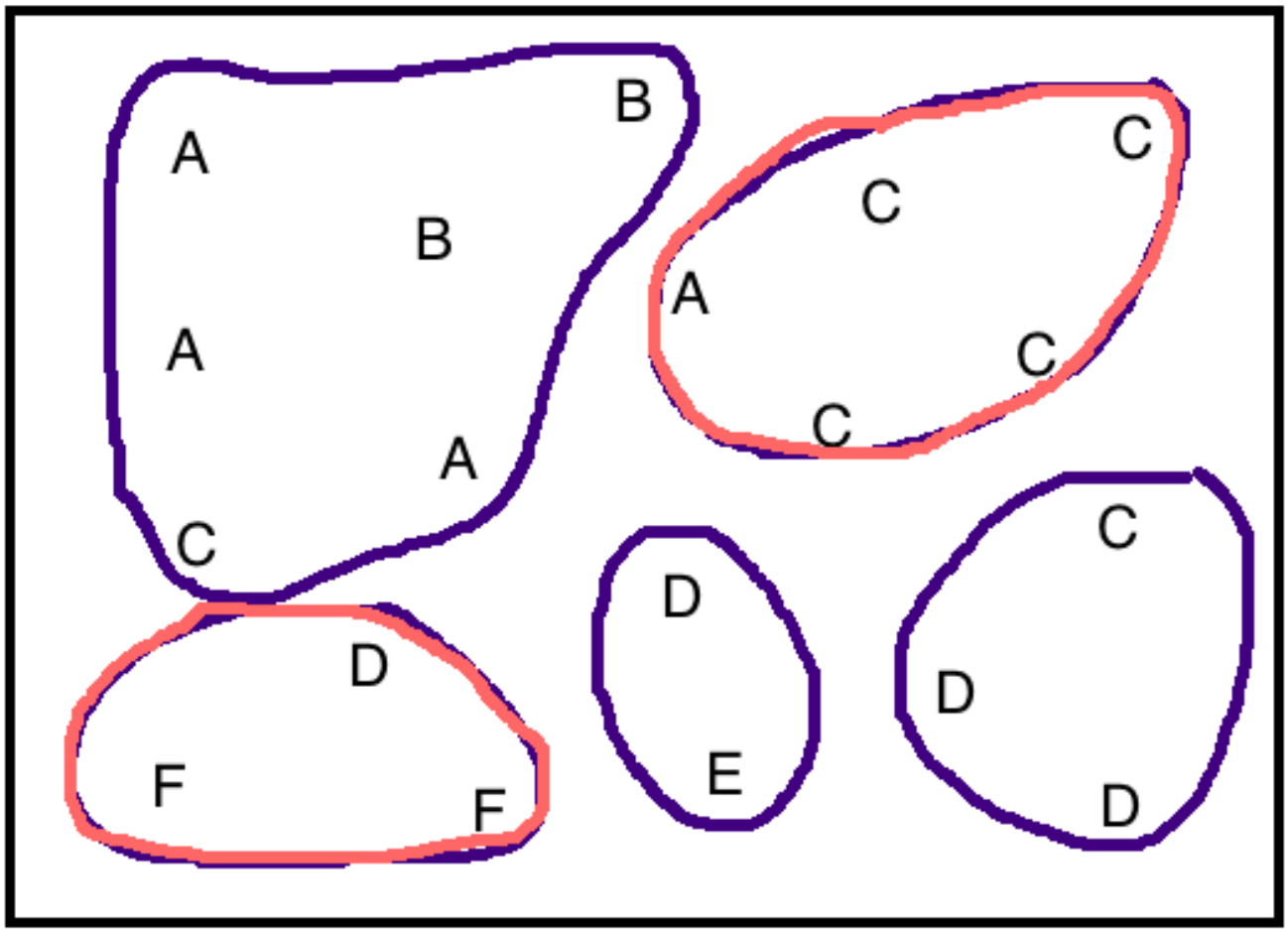
Clustered Random Sampling



Clustered Random Sampling



Clustered Random Sampling



Clustered Random Sampling

How can we calculate the sample mean?
First take mean within clusters and then across clusters!

$$\bar{y} = \frac{\sum_{a=1}^{A} \sum_{b=1}^{B} y_{ab}}{n}$$

- A - number of clusters
- B - number of elements with each cluster
- n - total sample size (the number of elements in all sampled clusters)

Clustered Random Sampling

How can we calculate the variance of the sample mean?
Now our random variation is in clusters, not elements.
Substitute sample standard deviation for between cluster standard deviation!

$$v(\bar{y}) = \frac{(1-f)}{A} s_A^2$$

$$s_A^2 = \frac{1}{A-1} \sum_{a=1}^{A} (\bar{y}_a - \bar{y})^2$$

- s_A^2 **- between cluster variance**

How does this compare to SRS?

We can estimate the design effect for our clustered random sample with:

$$d^2 = \frac{\left(\frac{1-f}{A}\right) s_A^2}{\left(\frac{1-f}{n}\right) s^2}$$

- $d^2 < 1 \rightarrow$ variance under clustered sampling is **lower**
- $d^2 > 1 \rightarrow$ variance under clustered sampling is **higher**
- $d^2 = 1 \rightarrow$ variance under clustered sampling is **the same**

How does this compare to SRS?

In reality, d^2 will always be > 1 in a clustered sample. I.e. the variance will be higher than what it would be under SRS. Why?

$$v(\bar{y}) = \frac{(1-f)}{A} s_A^2$$

As between cluster variance s_A^2 increases, so does our variance.

When between-cluster variance is high, within-cluster variance is likely to be low (cluster homogeneity).

What new information about the population do we obtain by adding to the sample one more element from the same cluster? Not much!

When Cluster?

When cluster?:

- Population has a clustered structure
- Unit-level sampling is expensive or not feasible
- Clusters are similar

Clustered Random Sampling

Advantages?

- Cost saving
- Better represents clustered structure

Disadvantages?

- Units tend to cluster for a reason
- Increased uncertainty if clusters differ from each other
- Complex to design (and possibly to administer)
- Analysis is more complex

One-Stage Clustered Random Sampling in R

```
# Sample
set.seed(1234)
cluster_sample <- cluster(apipop,clustername=c("cname"),
    size=10,method="srswor")
my_cluster <- getdata(apipop, cluster_sample)

# Tell R that this is survey data
pop_cluster_N <- length(table(apipop$cname))
my_cluster$pop_cluster_N <- rep(pop_cluster_N,
    dim(my_cluster)[1])
cluster_design <- svydesign(id=~cname, fpc=~pop_cluster_N,
    data=my_cluster)

# Estimate the population mean for the variable enroll
cluster_mean <- svymean(~api00, cluster_design)
# Mean of 665.23 and standard error of 13.306
```

Two-Stage Clustered Random Sampling

Same as clustered random sampling, but now we only sample some of the elements from the cluster.

- Group elements of sampling frame into clusters
- Randomly sample clusters using SRS
- Sample elements from within the sampled clusters proportionate to population size

Multi-Stage Designs

Can actually sample in many different stages using all of the methods describe.

Examples:

- Clustered at multiple levels
- Clustered and then stratified
- Stratified and then clustered

Just make sure to think about weighting!

Two-Stage Clustered Random Sampling in R

```
# Sample
set.seed(1234)
cluster2_sample <- mstage(apipop, stage=list("cluster", ""),
    varnames=list("cname"), size=list(10,rep(1,10)),
    method=list("srswor","srswor"))
my_cluster2 <- getdata(apipop, cluster2_sample)[[2]]

# Tell R that this is survey data
my_cluster2$pop_cluster_N <-
    rep(pop_cluster_N,dim(my_cluster2)[1])
cluster2_design <- svydesign(id=~cname, fpc=~pop_cluster_N,
    data=my_cluster2, nest=T)

# Estimate the population mean for the variable enroll
cluster2_mean <- svymean(~api00, cluster2_design)
# Mean of 686.2 and standard error of 21.59
# Look at how high our standard error is!
```

Weights: Clustered Random Sampling

We want to take a sample of m clusters. If we sample such that all clusters in the sample frame have equal probabilities of selection (SRS), we can calculate **cluster weights:**

$$\pi_j = \frac{m}{M}$$
$$W_j = \frac{M}{m}$$

- m - number of clusters sampled
- M - total number of Clusters in population

Weights: Clustered Random Sampling

Alternatively, you could cluster sample such that the probability that any cluster is sampled is proportional to the size of that cluster. In which we can calculate **cluster weights:**

$$\pi_j = \frac{m \times n_j}{N}$$

$$W_j = \frac{N}{m \times n_j}$$

- m - number of clusters sampled
- N - total population size
- n_j - sample size in cluster j

Weights: Clustered Random Sampling

But this only gives us the cluster-level weight. How do we get the individual-level weight?

$$W_i = W_j \times W_{i|j}$$

- W_j - cluster weight
- $W_{i|j}$ - individual weight for elements within cluster j.

This individual weight can be from a simple random sample or a stratified random sample.

Calculating Clustered Weights in R

```
# The pw variable store the design weight for each row
table(apiclus2$pw, apiclus2$stype)

# Can we figure out where they came from?
# First we need to calculate the cluster weight
# Number of clusters in population /
number of clusters in sample
apiclus2$cluster_weight <-
    length(table(apipop$dname))/length(table(apiclus2$dname))

# Second we need to calculate the school weight w/in cluster
# The pop number of schools in each cluster is in fpc2
apiclus2$school_weight <- NA
for(i in 1:length(apiclus2$school_weight)){
  apiclus2$school_weight[i] <-
    ifelse(apiclus2$fpc2[i]>=5, apiclus2$fpc2[i]/5, 1)
}

# Now we can calculate the total weight
apiclus2$weight <-
    apiclus2$cluster_weight*apiclus2$school_weight
# Voila!
```

Outline

Poststratification

Poststratification: the use of weights to assure that the sample totals equal some external total based on the target population. Essentially, it is "stratifying" after you have already run your sample.

- Aims to improve representativeness of sample
- Not always possible to stratify sample from beginning
- Can improve the efficiency of estimators and correct for differential nonresponse across strata
- BUT must have large enough sample in each poststratum

Example

	$\bar{y}$	Sample %	Population %
Men	180	20%	50%
Women	120	80%	50%

- Unweighted sample mean $= 180 \times .2 + 120 \times .8 = 132$
- Weighted sample mean $= 180 \times .5 + 120 \times .5 = 150$

Population Information

- Must have populations totals or percentages for all strata
- What we want representation over depends on population and research
- Must select appropriate population characteristics for sample

Poststratification Weights

Two ways to calculate:

1. By hand - works best when only one or two strata
2. Raking - allows for many strata

Note that we may want to trim weights that we think fall outside of a reasonable range!

Poststratification Weights

$$W_p = \frac{Population\%}{Sample\%}$$

Example

Strata	% of Population	% of Sample	W_p
Age 18-34: Male	.25	.15	
Age 18-34: Female	.22	.20	
Age 35-64: Male	.18	.25	
Age 35-64: Female	.20	.30	
Age 65+: Male	.05	.05	
Age 65+: Female	.10	.05	

Example

Strata	% of Population	% of Sample	W_p
Age 18-34: Male	.25	.15	1.67
Age 18-34: Female	.22	.20	
Age 35-64: Male	.18	.25	
Age 35-64: Female	.20	.30	
Age 65+: Male	.05	.05	
Age 65+: Female	.10	.05	

Example

Strata	% of Population	% of Sample	W_p
Age 18-34: Male	.25	.15	1.67
Age 18-34: Female	.22	.20	1.10
Age 35-64: Male	.18	.25	
Age 35-64: Female	.20	.30	
Age 65+: Male	.05	.05	
Age 65+: Female	.10	.05	

Example

Strata	% of Population	% of Sample	W_p
Age 18-34: Male	.25	.15	1.67
Age 18-34: Female	.22	.20	1.10
Age 35-64: Male	.18	.25	.72
Age 35-64: Female	.20	.30	
Age 65+: Male	.05	.05	
Age 65+: Female	.10	.05	

Example

Strata	% of Population	% of Sample	W_p
Age 18-34: Male	.25	.15	1.67
Age 18-34: Female	.22	.20	1.10
Age 35-64: Male	.18	.25	.72
Age 35-64: Female	.20	.30	.67
Age 65+: Male	.05	.05	
Age 65+: Female	.10	.05	

Example

Strata	% of Population	% of Sample	W_p
Age 18-34: Male	.25	.15	1.67
Age 18-34: Female	.22	.20	1.10
Age 35-64: Male	.18	.25	.72
Age 35-64: Female	.20	.30	.67
Age 65+: Male	.05	.05	1.00
Age 65+: Female	.10	.05	

Example

Strata	% of Population	% of Sample	W_p
Age 18-34: Male	.25	.15	1.67
Age 18-34: Female	.22	.20	1.10
Age 35-64: Male	.18	.25	.72
Age 35-64: Female	.20	.30	.67
Age 65+: Male	.05	.05	1.00
Age 65+: Female	.10	.05	2.00

Poststratification in R

Steps:

1. Specify your survey design with `svydesign()`
2. Store strata population totals in data frame
3. Poststratify the survey object to get poststratification adjusted weights with `postStratify()`
4. Trim the weights if too small/large

POSTSTRATIFICATION IN R

```
# 1. Specify survey design
clus_design <- svydesign(id=~dnum+snum, fpc=~fpc1+fpc2,
data=apiclus2)

# 2. Store the population totals
pop.types <- data.frame(stype=c("E", "H", "M"),
Freq=c(4421,755,1018))

# 3. Calculate poststratification adjusted weights
ps_design <- postStratify(clus_design, strata=~stype,
population=pop.types)

# 4. Run our models with this new design
clus.normal.glm <- svyglm(enroll ~ api00 + stype,
design = ps_design)
```

Raking

Raking: a model-based approach using known population totals to create poststratification weights so the marginal values of a table sum to the known totals.

- Repeated estimation of weights across strata until the weights converge.
- Allows multiple grouping variables to be used without knowing all cross-strata populations.

Raking in R

Steps:

1. Specify your survey design with `svydesign()`
2. Store strata population totals in data frames
3. Rake the survey object to get poststratification weights
4. Trim the weights if too small/large

Raking in R

```
# 0. Load Data
load(url("http://knutur.at/wsmt/R/RData/small.RData"))

# 1. Specify survey design
unweight_design <- svydesign(ids=~1, data=small)

# 2. Store the population totals
sex.dist <- data.frame(sex = c("M", "F"),
        Freq = c(45, 55))
edu.dist <- data.frame(edu = c("Lo", "Mid", "Hi"),
        Freq = c(30, 50, 20))

# 3. Calculate raking adjusted weights
rake_design <- rake(design = unweight_design,
        sample.margins = list(~sex, ~edu),
        population.margins = list(sex.dist, edu.dist))
summary(weights(rake_design))

# 4. Trim weights to be between .3 and 3
rake_design_trim <- trimWeights(rake_design, lower=0.3,
upper=3, strict=TRUE)
```

A Few R Notes

When you use the `postStratify()` or `rake()` commands, this will create a new survey design **with the adjusted weights**. I.e. It will have already combined your previous design/nonresponse weights with the poststratification weights.

Outline

Putting it All Together

We have now figured out how to calculate three different weights: W_i, W_r, W_p. We can only input one weight into our model. How do we get this?

$$W = W_i \times W_r \times W_p$$

To Weight or Not to Weight

Advantages?

- Improves representativeness
- Can help correct for coverage/sampling/nonresponse bias

Disadvantages?

- Usually increases your standard errors
- Complex to sort out

When Weight?

- Population Descriptive Statistics
  - Almost always! (Unless SRS with no nonresponse and perfect representation)
- Regression for model-based inference
  - Different probabilities of selection that are correlated with outcome (sampling bias)
    - But may solve just by controlling for strata
  - High nonresponse or coverage errors
  - Small sample size for subgroup
  - Correct for heteroskedasticity - weighted least squares

Good robustness test to try both with and without weights!