Algorithms of sampling with equal or unequal probabilities

Yves Tillé
University of Neuchâtel

Euskal Estatistika Erakundea
XXIII Seminario Internacional de Estadística
November 2010

Table of Contents

Population

General Concepts

General Ideas

- Three mains definitions.
  1. Supports or set of samples (example all the samples with replacement with fixed sample size n)
  2. Sampling design or multivariate discrete positive distribution.
  3. Sampling algorithms (applicable to any support and any design), ex: sequential algorithms.
- The application of a particular *sampling algorithm* on a *sampling design* defined on a *particular support* gives a sampling procedure.

Population

- Finite population, set of N units $\{u_1, \dots, u_k, \dots, u_N\}$.
- Each unit can be identified without ambiguity by a label.
- Let
$$U = \{1, \dots, k, \dots, N\}$$
be the set of these labels.

Variable of Interest

- The total $Y = \sum_{k \in U} y_k$,
- The population size $N = \sum_{k \in U} 1$,
- The mean $\overline{Y} = \frac{1}{N} \sum_{k \in U} y_k$,
- The variance $\sigma_y^2 = \frac{1}{N} \sum_{k \in U} \left(y_k - \overline{Y}\right)^2$.
- The corrected variance $V_y^2 = \frac{1}{N-1} \sum_{k \in U} \left(y_k - \overline{Y}\right)^2$.

Sample Without Replacement

- A sample without replacement is denoted by a column vector

$$\mathbf{s} = (s_1 \ \cdots \ s_k \ \cdots \ s_N)' \in \{0,1\}^N,$$

 where

$$s_k = \begin{cases} 1 & \text{if unit } k \text{ is in the sample} \\ 0 & \text{if unit } k \text{ is not in the sample,} \end{cases}$$

 for all $k \in U$.

- The sample size is $n(\mathbf{s}) = \sum_{k \in U} s_k$.

Sample With Replacement

- Samples with replacement,

$$\mathbf{s} = (s_1 \ \cdots \ s_k \ \cdots \ s_N)' \in \mathbb{N}^N,$$

  where $\mathbb{N} = \{0, 1, 2, 3, \dots\}$
  and s_k is the number of times that unit k is in the sample.

- The sample size is

$$n(\mathbf{s}) = \sum_{k \in U} s_k,$$

  and, in sampling with replacement, we can have $n(\mathbf{s}) > N$.

Support

Definition

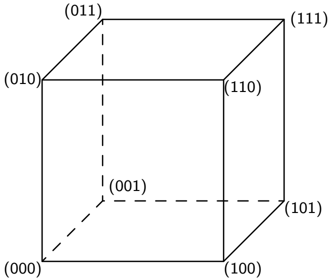
A support $\mathcal{Q}$ is a set of samples.

Definition

A support $\mathcal{Q}$ is said to be symmetric if, for any $\mathbf{s} \in \mathcal{Q}$, all the permutations of the coordinates of $\mathbf{s}$ are also in $\mathcal{Q}$.

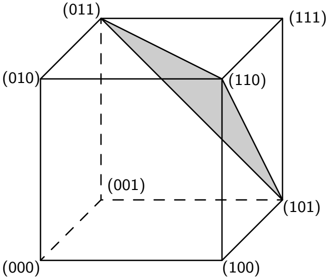
Particular symmetric supports 1

- The symmetric support without replacement: $\mathcal{S} = \{0,1\}^N$.
- Note that $\mathrm{card}(\mathcal{S}) = 2^N$.



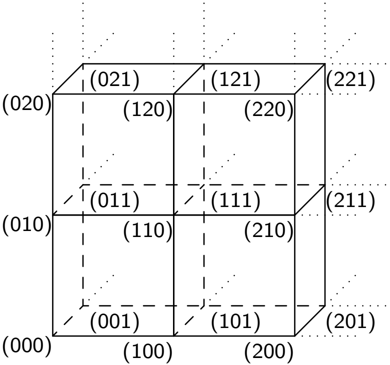
Particular symmetric supports 2

The symmetric support without replacement with fixed sample size
$\mathcal{S}_n = \left\{ \mathbf{s} \in \mathcal{S} \,\middle|\, \sum_{k \in U} s_k = n \right\}.$



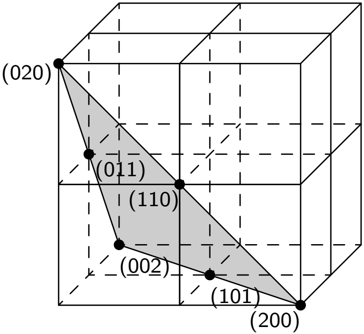
Particular symmetric supports 3

The symmetric support with replacement $\mathcal{R} = \mathbb{N}^N$,



Particular symmetric supports 4

The symmetric support with replacement of fixed size n
$\mathcal{R}_n = \left\{ \mathbf{s} \in \mathcal{R} \,\middle|\, \sum_{k \in U} s_k = n \right\}.$



Properties

1. $\mathcal{S}$, $\mathcal{S}_n$, $\mathcal{R}$, $\mathcal{R}_n$, are symmetric,
2. $\mathcal{S} \subset \mathcal{R}$,
3. The set $\{\mathcal{S}_0, \dots, \mathcal{S}_n, \dots, \mathcal{S}_N\}$ is a partition of $\mathcal{S}$,
4. The set $\{\mathcal{R}_0, \dots, \mathcal{R}_n, \dots, \mathcal{R}_N, \dots\}$ is an infinite partition of $\mathcal{R}$,
5. $\mathcal{S}_n \subset \mathcal{R}_n$, for all $n = 0, \dots, N$.

Sampling Design and Random Sample

Definition

A sampling design $p(.)$ on a support $\mathcal{Q}$ is a multivariate probability distribution on $\mathcal{Q}$; that is, $p(.)$ is a function from support $\mathcal{Q}$ to $]0,1]$ such that $p(\mathbf{s}) > 0$ for all $\mathbf{s} \in \mathcal{Q}$ and

$$\sum_{\mathbf{s} \in \mathcal{Q}} p(\mathbf{s}) = 1.$$

Remark

Because $\mathcal{S}$ can be viewed as the set of all the vertices of a hypercube, a sampling design without replacement is a probability measure on all these vertices.

Random Sample

Definition

A random sample $\mathbf{S} \in \mathbb{R}^N$ with the sampling design $p(.)$ is a random vector such that

$$\Pr(\mathbf{S} = \mathbf{s}) = p(\mathbf{s}), \text{ for all } \mathbf{s} \in \mathcal{Q},$$

where $\mathcal{Q}$ is the support of $p(.)$.

Expectation and variance

Definition

The expectation of a random sample $\mathbf{S}$ is

$$\boldsymbol{\mu} = \mathrm{E}(\mathbf{S}) = \sum_{\mathbf{s}\in\mathcal{Q}} p(\mathbf{s})\mathbf{s}.$$

The joint expectation

$$\mu_{k\ell} = \sum_{\mathbf{s}\in\mathcal{Q}} p(\mathbf{s}) s_k s_\ell.$$

The variance-covariance operator

$$\boldsymbol{\Sigma} = [\Sigma_{k\ell}] = \mathrm{var}(\mathbf{S}) = \sum_{\mathbf{s}\in\mathcal{Q}} p(\mathbf{s})(\mathbf{s}-\boldsymbol{\mu})(\mathbf{s}-\boldsymbol{\mu})' = [\mu_{k\ell} - \mu_k\mu_\ell].$$

Inclusion probabilities

Definition

The first-order inclusion probability is the probability that unit k is in the random sample

$$\pi_k = \Pr(S_k > 0) = \mathrm{E}[r(S_k)],$$

where $r(.)$ is the reduction function

$$r(S_k) = \begin{cases} 1 & \text{if } S_k > 0 \\ 0 & \text{if } S_k = 0 \end{cases}$$

$$\boldsymbol{\pi} = (\pi_1 \ \cdots \ \pi_k \ \cdots \ \pi_N)'.$$

Inclusion probabilities

Definition

The joint inclusion probability is the probability that unit k and ℓ are together in the random sample

$$\pi_{k\ell} = \Pr(S_k > 0 \text{ and } S_\ell > 0) = \mathrm{E}\left[r(S_k)r(S_\ell)\right],$$

with $\pi_{kk} = \pi_k, k \in U$.

Let $\mathbf{\Pi} = [\pi_{k\ell}]$ be the matrix of joint inclusion probabilities. Moreover, we define

$$\mathbf{\Delta} = \mathbf{\Pi} - \boldsymbol{\pi}\boldsymbol{\pi}'.$$

Inclusion probabilities

Result

$$\sum_{k \in U} \pi_k = \mathrm{E}\left\{ n[r(\mathbf{S})] \right\},$$

and

$$\sum_{k \in U} \Delta_{k\ell} = \mathrm{E}\left\{ n[r(\mathbf{S})] \left(r(S_\ell) - \pi_\ell \right) \right\}, \text{ for all } \ell \in U.$$

Moreover, if $\operatorname{var}\left\{ n[r(\mathbf{S})] \right\} = 0$ *then*

$$\sum_{k \in U} \Delta_{k\ell} = 0, \text{ for all } \ell \in U.$$

Computation of the Inclusion Probabilities

- Auxiliary variables $x_k > 0, k \in U$.
- First, compute the quantities

$$\frac{nx_k}{\sum_{\ell \in U} x_\ell}, \quad (1)$$

  $k = 1, \ldots, N$.
- For units for which these quantities are larger than 1, set $\pi_k = 1$. Next, the quantities are recalculated using (1) restricted to the remaining units.

Characteristic Function

The characteristic function $\phi(\mathbf{t})$ from $\mathbb{R}^N$ to $\mathbb{C}$ of a random sample $\mathbf{S}$ with sampling design $p(.)$ on $\mathcal{Q}$ is defined by

$$\phi_{\mathbf{S}}(\mathbf{t}) = \sum_{\mathbf{s}\in\mathcal{Q}} e^{i\mathbf{t}'\mathbf{s}} p(\mathbf{s}), \mathbf{t} \in \mathbb{R}^N, \tag{2}$$

where $i = \sqrt{-1}$, and $\mathbb{C}$ is the set of the complex numbers.

$$\phi'(0) = i\boldsymbol{\mu}, \qquad and \qquad \phi''(0) = -(\boldsymbol{\Sigma} + \boldsymbol{\mu}\boldsymbol{\mu}').$$

Hansen-Hurwitz Estimator

The Hansen-Hurwitz estimator(see Hansen and Hurwitz, 1943) of Y is defined by

$$\widehat{Y}_{HH} = \sum_{k \in U} \frac{S_k y_k}{\mu_k},$$

where $\mu_k = \mathrm{E}(S_k), k \in U$.

Result

If $\mu_k > 0$, for all $k \in U$, then $\widehat{Y}_{HH}$ is an unbiased estimator of Y.

Horvitz-Thompson Estimator

The Horvitz-Thompson estimator(see Horvitz and Thompson, 1952) is defined by

$$\widehat{Y}_{HT} = \sum_{k \in U} \frac{r(S_k) y_k}{\pi_k},$$

where

$$r(S_k) = \begin{cases} 0 & \text{if } S_k = 0 \\ 1 & \text{if } S_k > 0. \end{cases}$$

Population

Sampling Algorithms

Sampling Algorithm

Definition

A sampling algorithm is a procedure allowing the selection of a random sample.

An algorithm must be a shortcut that avoid the combinatory explosion.

Enumerative Algorithms 1

Algorithm Enumerative algorithm

1. First, construct a list $\{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_j, \dots, \mathbf{s}_J\}$ of all possible samples with their probabilities.
2. Next, generate a random variable u with a uniform distribution in [0,1].
3. Finally, select the sample s_j such that $\sum_{i=1}^{j-1} p(\mathbf{s}_i) \leq u < \sum_{i=1}^{j} p(\mathbf{s}_i)$.

Enumerative Algorithms 2

Table: Sizes of symmetric supports

Support $\mathcal{Q}$	card($\mathcal{Q}$)	$N = 100, n = 10$	$N = 300, n = 30$
$\mathcal{R}$	∞	–	–
$\mathcal{R}_n$	$\binom{N+n-1}{n}$	5.1541×10^{13}	3.8254×10^{42}
$\mathcal{S}$	2^N	1.2677×10^{30}	2.0370×10^{90}
$\mathcal{S}_n$	$\binom{N}{n}$	1.7310×10^{13}	1.7319×10^{41}

Sequential Algorithms

A sequential procedure is a method that is applied to a list of units sorted according to a particular order denoted $1, \dots, k, \dots, N$.

Definition

A sampling procedure is said to be weakly sequential if at step $k = 1, \dots, N$ of the procedure, the decision concerning the number of times that unit k is in the sample is definitively taken.

Definition

A sampling procedure is said to be strictly sequential if it is weakly sequential and if the decision concerning unit k does not depend on the units that are after k on the list.

Standard Sequential Algorithms

Algorithm Standard sequential procedure

---

1. Let $p(\mathbf{s})$ be the sampling design and $\mathcal{Q}$ the support. First, define

$$q_1(s_1) = \Pr(S_1 = s_1) = \sum_{\mathbf{s} \in \mathcal{Q} | S_1 = s_1} p(\mathbf{s}), s_1 = 0, 1, 2, \dots$$

2. Select the first unit s_1 times according to the distribution $q_1(s_1)$.
3. For $k = 2, \dots, N$ do
   1. Compute

$$\begin{aligned} q_k(s_k) &= \Pr(S_k = s_k | S_{k-1} = s_{k-1}, \dots, S_1 = s_1) \\ &= \frac{\sum_{\mathbf{s} \in \mathcal{Q} | S_k = s_k, S_{k-1} = s_{k-1}, \dots, S_1 = s_1} p(\mathbf{s})}{\sum_{\mathbf{s} \in \mathcal{Q} | S_{k-1} = s_{k-1}, \dots, S_1 = s_1} p(\mathbf{s})}, s_k =, 0, 1, 2, \dots \end{aligned}$$

   2. Select the kth unit s_k times according to the distribution $q_k(s_k)$;

EndFor.

Draw by draw Algorithms

The draw by draw algorithms are restricted to designs with fixed sample size. We refer to the following definition.

Definition

A sampling design of fixed sample size n is said to be draw by draw if, at each one of the n steps of the procedure, a unit is definitively selected in the sample.

Standard Draw by Draw Algorithm

Algorithm Standard draw by draw algorithm

1. Let $p(\mathbf{s})$ be a sampling design and $\mathcal{Q} \subset \mathcal{R}_n$ the support. First, define $p^{(0)}(\mathbf{s}) = p(\mathbf{s})$ and $\mathcal{Q}(0) = \mathcal{Q}$. Define also $\mathbf{b}(0)$ as the null vector of $\mathbb{R}^N$.
2. FOR $t = 0, \dots, n-1$ DO
   1. Compute $\boldsymbol{\nu}(t) = \sum_{\mathbf{s} \in \mathcal{Q}(t)} \mathbf{s} p^{(t)}(\mathbf{s})$;
   2. Select randomly one unit from U with probabilities $q_k(t)$, where
      $$q_k(t) = \frac{\nu_k(t)}{\sum_{\ell \in U} \nu_\ell(t)} = \frac{\nu_k(t)}{n-t}, k \in U;$$
      The selected unit is denoted j;
   3. Define $\mathbf{a}_j = (0 \cdots 0 \underbrace{1}_{j\text{th}} 0 \cdots 0)$; Execute $\mathbf{b}(t+1) = \mathbf{b}(t) + \mathbf{a}_j$;
   4. Define $\mathcal{Q}(t+1) = \left\{\tilde{\mathbf{s}} = \mathbf{s} - \mathbf{a}_j, \text{ for all } \mathbf{s} \in \mathcal{Q}(t) \text{ such that } s_j > 0\right\}$;
   5. Define, for all $\tilde{\mathbf{s}} \in \mathcal{Q}(t+1)$, $p^{(t+1)}(\tilde{\mathbf{s}}) = \frac{s_j p^{(t)}(\mathbf{s})}{\sum_{\mathbf{s} \in \mathcal{Q}(t)} s_j p^{(t)}(\mathbf{s})}$, where $\mathbf{s} = \tilde{\mathbf{s}} + \mathbf{a}_j$;
3. The selected sample is $\mathbf{b}(n)$.

Standard Draw by draw Algorithm (without replacement)

Algorithm Standard draw by draw algorithm for sampling without replacement

1. Let $p(\mathbf{s})$ be a sampling design and $\mathcal{Q} \in \mathcal{S}$ the support.
2. Define $\mathbf{b} = (b_k) = \mathbf{0} \in \mathbb{R}^N$.
3. FOR $t = 0, \ldots, n-1$ DO
   select a unit from U with probability

$$q_k = \begin{cases} \dfrac{1}{n-t} \mathrm{E}\left(S_k | S_i = 1 \text{ for all } i \text{ such that } b_i = 1\right) & \text{if } b_k = 0 \\ 0 & \text{if } b_k = 1; \end{cases}$$

   IF unit j is selected, THEN $b_j = 1$;

Other Algorithms

- Eliminatory algorithms (Chao, Tillé)
- Splitting methods
- Rejective algorithms
- Systematic algorithms
- Others algorithms (Sampford)

Population

Simple Random Sampling

Simple Random Sampling

Definition

A sampling design $p_{\text{SIMPLE}}(., \theta, \mathcal{Q})$ of parameter $\theta \in \mathbb{R}_+^*$ on a support $\mathcal{Q}$ is said to be simple, if

(i) Its sampling design can be written

$$p_{\text{SIMPLE}}(\mathbf{s}, \theta, \mathcal{Q}) = \frac{\theta^{n(\mathbf{s})} \prod_{k \in U} 1/s_k!}{\sum_{\mathbf{s} \in \mathcal{Q}} \theta^{n(\mathbf{s})} \prod_{k \in U} 1/s_k!}, \quad \text{for all } \mathbf{s} \in \mathcal{Q}.$$

(ii) Its support $\mathcal{Q}$ is symmetric (see Definition 2, page 9).

Simple Random Sampling

- Support $\mathcal{S}$ Bernoulli sampling
- Support $\mathcal{S}_n$ Simple Random Sampling Without Replacement
- Support $\mathcal{R}$ Bernoulli sampling With replacement
- Support $\mathcal{R}_n$ Simple Random Sampling With Replacement

Links between simple designs

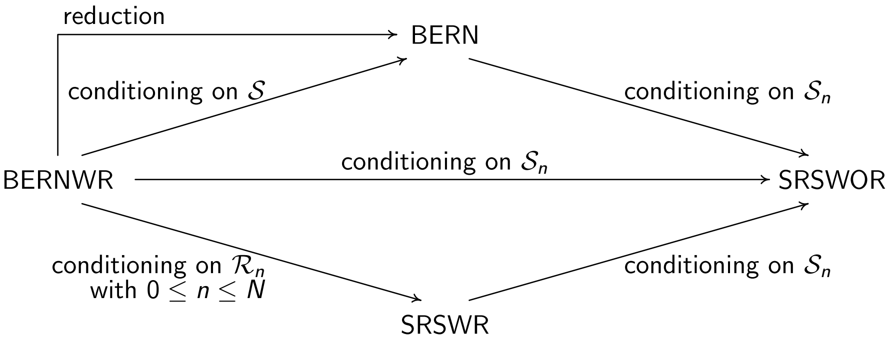


Figure: Links between the main simple sampling designs

Main simple random sampling designs

Notation	BERNWR	SRSWR	BERN	SRSWOR
$p(\mathbf{s})$	$\frac{\mu^{n(\mathbf{s})}}{e^{N\mu}} \prod_{k\in U} \frac{1}{s_k!}$	$\frac{n!}{N^n} \prod_{k\in U} \frac{1}{s_k!}$	$\pi^{n(\mathbf{s})}(1-\pi)^{N-n(\mathbf{s})}$	$\binom{N}{n}^{-1}$
$\mathcal{Q}$	$\mathcal{R}$	$\mathcal{R}_n$	$\mathcal{S}$	$\mathcal{S}_n$
$\phi(\mathbf{t})$	$\exp\left\{\mu \sum_{k\in U}(e^{it_k}-1)\right\}$	$\left(\frac{1}{N}\sum_{k\in U} e^{it_k}\right)^n$	$\prod_{k\in U}\left\{1+\pi\left(e^{it_k}-1\right)\right\}$	$\binom{N}{n}^{-1} \sum_{\mathbf{s}\in\mathcal{S}_n} e^{i\mathbf{t}'\mathbf{s}}$
WOR/WR	with repl.	with repl.	without repl.	without repl.
$n(\mathbf{S})$	random	fixed	random	fixed
μ_k	μ	$\frac{n}{N}$	π	$\frac{n}{N}$
π_k	$1-e^{-\mu}$	$1-\left(\frac{N-1}{N}\right)^n$	π	$\frac{n}{N}$

Sequential procedure on Bernoulli sampling

Algorithm Bernoulli sampling without replacement

DEFINITION k : INTEGER;
FOR $k = 1, \ldots, N$ DO with probability π select unit k; ENDFOR.

Draw by draw procedure for SRSWOR

Algorithm Draw by draw procedure for SRSWOR

DEFINITION j : INTEGER;
FOR $t = 0, \dots, n-1$ DO
select a unit k from the population with probability

$$q_k = \begin{cases} \frac{1}{N-t} & \text{if } k \text{ is not already selected} \\ 0 & \text{if } k \text{ is already selected;} \end{cases}$$

ENDFOR.

Sequential procedure for SRSWOR

Fan et al. (1962)

Algorithm Selection-rejection procedure for SRSWOR

DEFINITION k, j : INTEGER;
$j = 0$;
FOR $k = 1, \dots, N$ DO
with probability $\dfrac{n-j}{N-(k-1)}$ THEN | select unit k; $j = j + 1$;
ENDFOR.

Draw by Draw Procedure for SRSWR

Algorithm Draw by Draw Procedure for SRSWR

DEFINITION j : INTEGER;
FOR $j = 1, \dots, n$ DO
    a unit is selected with equal probability $1/N$ from the population U;
ENDFOR.

Sequential Procedure for SRSWR

Algorithm Sequential procedure for SRSWR

DEFINITION k, j : INTEGER;
$j = 0$;
FOR $k = 1, \ldots, N$ DO
select the kth unit s_k times according to the binomial distribution

$$\mathcal{B}\left(n - \sum_{i=1}^{k-1} s_i, \frac{1}{N-k+1}\right);$$

ENDFOR.

Unequal Probability Sampling

Unequal Probability Sampling

Why the problem is complex? False method

Selection of 2 units with unequal probability

$$p_k = \frac{x_k}{\sum_{\ell \in U} x_\ell}, k \in U.$$

The generalization is the following:

- At the first step, select a unit with unequal probability $p_k, k \in U$.
- The selected unit is denoted j.
- The selected unit is removed from U.
- Next we compute

$$p_k^j = \frac{p_k}{1 - p_j}, k \in U \backslash \{j\}.$$

Why the problem is complex? 2

Select again a unit with unequal probabilities $p_k^j, k \in U$, amongst the $N - 1$ remaining units, and so on.
This method is wrong.
We can see it by taking $n = 2$.

Why the problem is complex? 3

In this case,

$$\begin{aligned}
\Pr(k \in S) &= \Pr(k \text{ be selected at the first step }) \\
&\quad + \Pr(k \text{ be selected at the second step }) \\
&= p_k + \sum_{\substack{j \in U \\ j \neq k}} p_j p_k^j \\
&= p_k \left(1 + \sum_{\substack{j \in U \\ j \neq k}} \frac{p_j}{1 - p_j} \right). \qquad (3)
\end{aligned}$$

We should have $\pi_k = 2p_k, k \in U$.

Why the problem is complex? 4

We could use modified values p_k^* for the p_k in such a way that the inclusion probabilities is equal to π_k.
In the case where $n = 2$, we should have p_k^* such that

$$p_k^* \left(1 + \sum_{\substack{j \in U \\ j \neq k}} \frac{p_j^*}{1 - p_j^*} \right) = \pi_k, k \in U.$$

This method is known as the Nairin procedure (see also Horvitz and Thompson, 1952; Yates and Grundy, 1953; Brewer and Hanif, 1983, p.25)

Systematic sampling 1

Madow (1949)
Fixed sample size and exact method.
We have $0 < \pi_k < 1, k \in U$ with

$$\sum_{k \in U} \pi_k = n.$$

Define $V_k = \sum_{\ell=1}^{k} \pi_\ell,$ for all $k \in U,$ with $V_o = 0$. A uniform random number is generated in [0, 1].

- the first unit selected k_1 is such that $V_{k_1 - 1} \leq u < V_{k_1}$,
- the second unit selected is such that $V_{k_2 - 1} \leq u + 1 < V_{k_2}$ and
- the jth unit selected is such that $V_{k_j - 1} \leq u + j - 1 < V_{k_j}$.

Systematic sampling 2

Example

Suppose that $N = 6$ and $n = 3$.

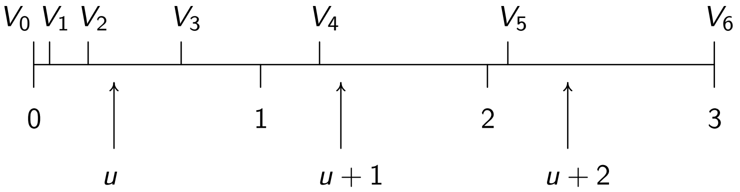
k	0	1	2	3	4	5	6	Total
π_k	0	0.07	0.17	0.41	0.61	0.83	0.91	3
V_k	0	0.07	0.24	0.65	1.26	2.09	3	

Systematic sampling 3

Suppose also that the value taken by the uniform random number is $u = 0.354$. The rules of selections are:

- Because $V_2 \leq u < V_3$, unit 3 is selected;
- Because $V_4 \leq u < V_5$, unit 5 is selected;
- Because $V_5 \leq u < V_6$, unit 6 is selected.

The sample selected is thus $\mathbf{s} = (0, 0, 1, 0, 1, 1)$.



Systematic sampling 4

$$\mathbf{\Pi} = \begin{pmatrix} 0.07 & 0 & 0 & 0.07 & 0.07 & 0 \\ 0 & 0.17 & 0 & 0.17 & 0.02 & 0.15 \\ 0 & 0 & 0.41 & 0.02 & 0.39 & 0.41 \\ 0.07 & 0.17 & 0.02 & 0.61 & 0.44 & 0.52 \\ 0.07 & 0.02 & 0.39 & 0.44 & 0.83 & 0.74 \\ 0 & 0.15 & 0.41 & 0.52 & 0.74 & 0.91 \end{pmatrix}.$$

Systematic sampling 5

Algorithm Systematic sampling

DEFINITION a, b, u real; k INTEGER;
$u = \mathcal{U}[0, 1[$;
$a = -u$;
FOR $k = 1, \dots, N$ DO
 $b = a$;
 $a = a + \pi_k$;
 IF $\lfloor a \rfloor \neq \lfloor b \rfloor$ THEN select k ENDIF;
ENDFOR.

Systematic sampling 6

Problem: most of the joint inclusion probabilities are equal to zero.
Matrix of the joint inclusion probabilities:

$$\begin{bmatrix} - & 0 & 0.2 & 0.2 & 0 & 0 \\ 0 & - & 0.5 & 0.2 & 0.4 & 0.3 \\ 0.2 & 0.5 & - & 0.3 & 0.4 & 0.2 \\ 0.2 & 0.2 & 0.3 & - & 0 & 0.3 \\ 0 & 0.4 & 0.4 & 0 & - & 0 \\ 0 & 0.3 & 0.2 & 0.3 & 0 & - \end{bmatrix}$$

Systematic sampling 7

- The sampling design depends on the order of the population.
- When the variable of interest depends on the order of the file, the variance is reduced.
- **Random systematic sampling:** The file is sorted randomly before applying random systematic sampling.

Exponential family

Definition

A sampling design $p_{\mathrm{EXP}}(.)$ on a support $\mathcal{Q}$ is said to be exponential if it can be written

$$p_{\mathrm{EXP}}(\mathbf{s}, \boldsymbol{\lambda}, \mathcal{Q}) = g(\mathbf{s}) \exp\left[\boldsymbol{\lambda}'\mathbf{s} - \alpha(\boldsymbol{\lambda}, \mathcal{Q})\right],$$

where $\boldsymbol{\lambda} \in \mathbb{R}^N$ is the parameter,

$$g(\mathbf{s}) = \prod_{k \in U} \frac{1}{s_k!},$$

and $\alpha(\boldsymbol{\lambda}, \mathcal{Q})$ is called the normalizing constant and is given by

$$\alpha(\boldsymbol{\lambda}, \mathcal{Q}) = \log \sum_{\mathbf{s} \in \mathcal{Q}} g(\mathbf{s}) \exp \boldsymbol{\lambda}'\mathbf{s}.$$

Expectation

- The expectation
$$\boldsymbol{\mu}(\boldsymbol{\lambda}) = \sum_{\mathbf{s}\in\mathcal{Q}} \mathbf{s}\, p_{\mathrm{EXP}}(\mathbf{s}, \boldsymbol{\lambda}, \mathcal{Q})$$
- The function $\boldsymbol{\mu}(\boldsymbol{\lambda})$ is bijective. (fundamental result on exponential families).
- The most important in an exponential family is its parameter and not $\boldsymbol{\mu}$ or $\boldsymbol{\pi}$.

Main exponential designs

Notation	POISSWR	MULTI	POISSWOR	CPS
$p(\mathbf{s})$	$\prod_{k\in U} \frac{\mu_k^{s_k} e^{-\mu_k}}{s_k!}$	$\frac{n!}{n^n} \prod_{k\in U} \frac{\mu_k^{s_k}}{s_k!}$	$\prod_{k\in U} \left[\pi_k^{s_k}(1-\pi_k)^{1-s_k}\right]$	$\sum_{\mathbf{s}\in\mathcal{S}_n} \exp[\lambda'\mathbf{s} - \alpha(\lambda, \mathcal{S}_n)]$
$\mathcal{Q}$	$\mathcal{R}$	$\mathcal{R}_n$	$\mathcal{S}$	$\mathcal{S}_n$
$\alpha(\lambda, Q)$	$\sum_{k\in U} \exp \lambda_k$	$\log \frac{1}{n!} \left(\sum_{k\in U} \exp \lambda_k\right)^n$	$\log \prod_{k\in U} (1 + \exp \lambda_k)$	difficult
$\phi(\mathbf{t})$	$\exp \sum_{k\in U} \mu_k(e^{it_k} - 1)$	$\left(\frac{1}{n} \sum_{k\in U} \mu_k \exp it_k\right)^n$	$\prod_{k\in U} \{1 + \pi_k (\exp it_k - 1)\}$	not reducible
WOR/WR	with repl.	with repl.	without repl.	without repl.
$n(\mathbf{S})$	random	fixed	random	fixed
μ_k	$\mu_k = \exp \lambda_k$	$\mu_k = \frac{n \exp \lambda_k}{\sum_{k\in U} \exp \lambda_k}$	$\pi_k = \frac{\exp \lambda_k}{1+\exp \lambda_k}$	$\pi_k(\lambda, \mathcal{S}_n)$ difficult
π_k	$1 - e^{-\mu_k}$	$1 - (1 - \mu_k/n)^n$	π_k	$\pi_k(\lambda, \mathcal{S}_n)$

Sequential procedure for multinomial design

Algorithm Sequential procedure for multinomial design

DEFINITION k : INTEGER;
FOR $k = 1, \ldots, N$ DO
  select the kth unit s_k times according to the binomial distribution

$$\mathcal{B}\left(n - \sum_{\ell=1}^{k-1} s_\ell, \frac{\mu_k}{n - \sum_{\ell=1}^{k-1} \mu_\ell}\right);$$

ENDFOR.

Draw by draw procedure for multinomial design

Algorithm Draw by draw procedure for multinomial design

DEFINITION j : INTEGER;
FOR $j = 1, \ldots, n$ DO
a unit is selected with probability μ_k / n from the population U;
ENDFOR.

Sequential procedure for POISSWOR

Algorithm Sequential procedure for POISSWOR

DEFINITION k : INTEGER;
FOR $k = 1, \ldots, N$, DO select the kth unit with probability π_k;
ENDFOR.

Conditional Poisson Sampling (CPS)

- CPS = Exponential design on $\mathcal{S}_n$ (or maximum entropy design)
- Chen et al. (1994) and Deville (2000)
- $p_{\mathrm{CPS}}(\mathbf{s}, \boldsymbol{\lambda}, n) = p_{\mathrm{EXP}}(\mathbf{s}, \boldsymbol{\lambda}, \mathcal{S}_n) = \dfrac{\exp \boldsymbol{\lambda}'\mathbf{s}}{\sum_{\mathbf{s}\in\mathcal{S}_n} \exp \boldsymbol{\lambda}'\mathbf{s}}$
- The relation between $\boldsymbol{\lambda}$ and $\boldsymbol{\pi}$ is complex, but there exists the recursive relation:

$$\pi_k(\boldsymbol{\lambda}, \mathcal{S}_n) = n \frac{\exp \lambda_k \left[1 - \pi_k(\boldsymbol{\lambda}, \mathcal{S}_{n-1})\right]}{\sum_{\ell\in U} \exp \lambda_\ell \left[1 - \pi_\ell(\boldsymbol{\lambda}, \mathcal{S}_{n-1})\right]} (\text{with } \pi_\ell(\boldsymbol{\lambda}, \mathcal{S}_0) = 0)$$

- For obtaining $\boldsymbol{\lambda}$ from $\boldsymbol{\pi}$, the Newton method can be used.

Rejective procedure

- For example, select poisson samples until obtaining a fixed sample size. $p_{\mathrm{CPS}}(\mathbf{s}, \boldsymbol{\lambda}, n) = p_{\mathrm{EXP}}(\mathbf{s}, \boldsymbol{\lambda}, \mathcal{S}_n) = \dfrac{p_{\mathrm{EXP}}(\mathbf{s}, \boldsymbol{\lambda}, \mathcal{S})}{\sum_{\mathbf{s} \in \mathcal{S}_n} p_{\mathrm{EXP}}(\mathbf{s}, \boldsymbol{\lambda}, \mathcal{S})}$
  $p_{\mathrm{EXP}}(\mathbf{s}, \boldsymbol{\lambda}, \mathcal{S})$ is a Poisson design
  $p_{\mathrm{EXP}}(\mathbf{s}, \boldsymbol{\lambda}, \mathcal{S}_n)$ is a conditional Poisson design
- Warning: the use of a rejective procedure changes the inclusion probabilities.
- The parameter of the exponential design remains the same.

Idea of implementations: Rejective procedure

- The π_k of the CPS are given.
- Compute $\boldsymbol{\lambda}$ from the π_k by the Newton method.
- Compute the inclusion probabilities of the Poisson design

$$\tilde{\pi}_k = \exp(\lambda_k + C)/[1 + \exp(\lambda_k + C)].$$

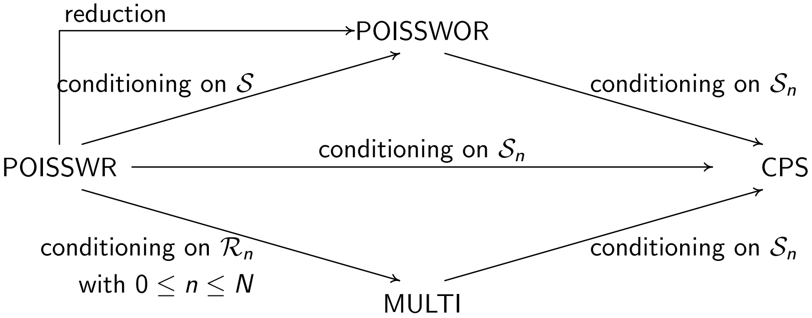
- Select Poisson samples until obtaining the good sample size n.

Implementation of CPS

- Sequential procedure
- Draw by draw procedure
- Poisson rejective procedure
- Multinomial rejective procedure

For all these procedures, $\boldsymbol{\lambda}$ must first be computed from $\boldsymbol{\pi}$. Next, the implementation becomes relatively simple.

Link between the exponential methods



The splitting method

Splitting Method

Basic splitting method

Deville and Tillé (1998)

π_k is split into two parts $\pi_k^{(1)}$ and $\pi_k^{(2)}$ that must satisfy:

$$\pi_k = \lambda \pi_k^{(1)} + (1-\lambda)\pi_k^{(2)}; \tag{4}$$

$$0 \leq \pi_k^{(1)} \leq 1 \text{ and } 0 \leq \pi_k^{(2)} \leq 1, \tag{5}$$

$$\sum_{k\in U} \pi_k^{(1)} = \sum_{k\in U} \pi_k^{(2)} = n, \tag{6}$$

where λ can be chosen freely provided that $0 < \lambda < 1$. The method consists of drawing n units with unequal probabilities

$$\begin{cases} \pi_k^{(1)}, k \in U, & \text{with a probability } \lambda \\ \pi_k^{(2)}, k \in U, & \text{with a probability } 1-\lambda. \end{cases}$$

Basic splitting method

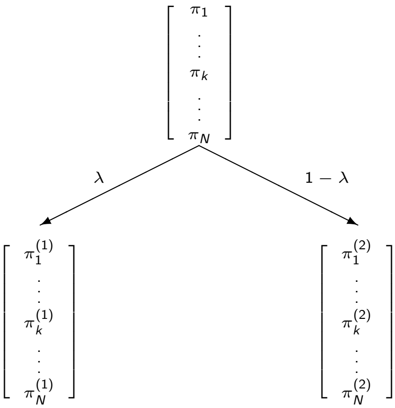


Figure: Splitting into two parts

Splitting method into M parts

Construct the $\pi_k^{(j)}$ and the λ_j in such a way that

$$\sum_{j=1}^{M} \lambda_j = 1,$$

$$0 \leq \lambda_j \leq 1 \qquad (j = 1, ..., M),$$

$$\sum_{j=1}^{M} \lambda_j \pi_k^{(j)} = \pi_k,$$

$$0 \leq \pi_k^{(j)} \leq 1 \qquad (k \in U, j = 1, ..., M),$$

$$\sum_{k \in U} \pi_k^{(j)} = n \qquad (j = 1, ..., M).$$

Splitting method into M parts

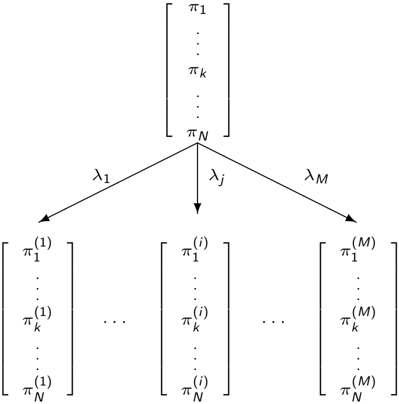


Figure: Splitting into M parts

Minimal Support Design

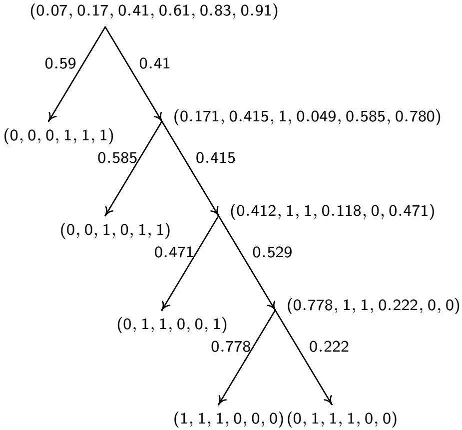
Denote by $\pi_{(1)}, ..., \pi_{(k)}, ..., \pi_{(N)}$ the ordered inclusion probabilities. Next, define

$$\lambda = \min\{1 - \pi_{(N-n)}, \pi_{(N-n+1)}\},$$

$$\pi_{(k)}^{(1)} = \begin{cases} 0 & \text{if } k \leq N-n \\ 1 & \text{if } k > N-n, \end{cases}$$

$$\pi_{(k)}^{(2)} = \begin{cases} \dfrac{\pi_{(k)}}{1-\lambda} & \text{if } k \leq N-n \\ \dfrac{\pi_{(k)} - \lambda}{1-\lambda} & \text{if } k > N-n. \end{cases}$$

Example: Splitting tree for the minimal support design



Splitting into simple random sampling

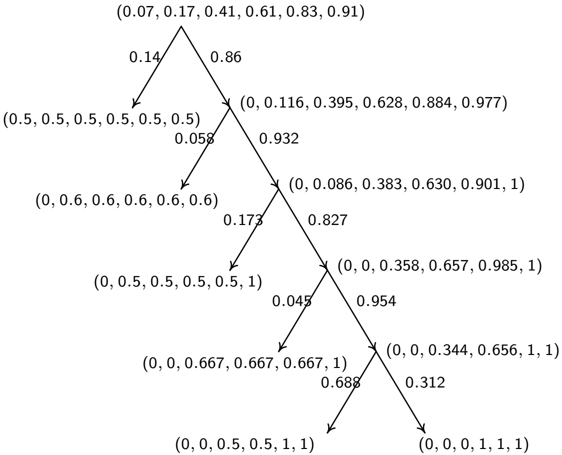
$$\lambda = \min \left\{ \pi_{(1)} \frac{N}{n}, \frac{N}{N-n}(1 - \pi_{(N)}) \right\}, \tag{7}$$

and compute, for $k \in U$,

$$\pi_{(k)}^{(1)} = \frac{n}{N}, \pi_{(k)}^{(2)} = \frac{\pi_k - \lambda \frac{n}{N}}{1 - \lambda}.$$

If $\lambda = \pi_{(1)} N/n$, then $\pi_{(1)}^{(2)} = 0$; if $\lambda = (1 - \pi_{(N)})N/(N-n)$, then $\pi_{(N)}^{(2)} = 1$. At the next step, the problem is thus reduced to a selection of a sample of size $n-1$ or n from a population of size $N-1$. In at most $N-1$ steps, the problem is solved.

Splitting tree for splitting into simple random sampling



Pivotal Method

At each step, only two unit are modifies i and j.
Two cases: If $\pi_i + \pi_j > 1$, then

$$\lambda = \frac{1-\pi_j}{2-\pi_i-\pi_j},$$

$$\pi_k^{(1)} = \begin{cases} \pi_k & k \in U \backslash \{i,j\} \\ 1 & k = i \\ \pi_i + \pi_j - 1 & k = j, \end{cases}$$

$$\pi_k^{(2)} = \begin{cases} \pi_k & k \in U \backslash \{i,j\} \\ \pi_i + \pi_j - 1 & k = i \\ 1 & k = j. \end{cases}$$

Pivotal Method

If $\pi_i + \pi_j < 1$, then

$$\lambda = \frac{\pi_i}{\pi_i + \pi_j},$$

$$\pi_k^{(1)} = \begin{cases} \pi_k & k \in U \backslash \{i,j\} \\ \pi_i + \pi_j & k = i \\ 0 & k = j, \end{cases} \quad \text{and } \pi_k^{(2)} = \begin{cases} \pi_k & k \in U \backslash \{i,j\} \\ 0 & k = i \\ \pi_i + \pi_j & k = j. \end{cases}$$

Brewer's Method

Brewer and Hanif (1983, p.26)
Brewer (1975)
draw by draw procedure

$$\lambda_j \quad = \quad \left\{ \sum_{z=1}^{N} \frac{\pi_z(n - \pi_z)}{1 - \pi_z} \right\}^{-1} \frac{\pi_j(n - \pi_j)}{1 - \pi_j}.$$

Next, we compute

$$\pi_k^{(j)} = \begin{cases} \dfrac{\pi_k(n-1)}{n - \pi_j} & \text{if } k \neq j \\ 1 & \text{if } k = j. \end{cases}$$

Brewer's Method

The validity derives from the following result:

Theorem

$$\sum_{j=1}^{N} \lambda_j \pi_k^{(j)} = \pi_k,$$

for all $k = 1, \ldots, N$,

References

Brewer, K. R. W. (1975). A simple procedure for πpswor. *Australian Journal of Statistics*, 17:166–172.

Brewer, K. R. W. and Hanif, M. (1983). *Sampling with Unequal Probabilities*. Springer, New York.

Chen, S. X., Dempster, A. P., and Liu, J. S. (1994). Weighted finite population sampling to maximize entropy. *Biometrika*, 81:457–469.

Deville, J.-C. (2000). Note sur l'algorithme de Chen, Dempster et Liu. Technical report, CREST-ENSAI, Rennes.

Deville, J.-C. and Tillé, Y. (1998). Unequal probability sampling without replacement through a splitting method. *Biometrika*, 85:89–101.

Fan, C. T., Muller, M. E., and Rezucha, I. (1962). Development of sampling plans by using sequential (item by item) selection techniques and digital computer. *Journal of the American Statistical Association*, 57:387–402.

Hansen, M. H. and Hurwitz, W. N. (1943). On the theory of sampling from finite populations. *Annals of Mathematical Statistics*, 14:333–362.

Horvitz, D. G. and Thompson, D. J. (1952). A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47:663–685.

Madow, W. G. (1949). On the theory of systematic sampling, II. *Annals of Mathematical Statistics*, 20:333–354.

Tillé, Y. (2006). *Sampling Agorithms*. Springer, New York.

Yates, F. and Grundy, P. M. (1953). Selection without replacement from within strata with probability proportional to size. *Journal of the Royal Statistical Society*, B15:235–261.