149

WP : 149

Working Paper

विद्याविनियोगाद्विकासः
IIM
AHMEDABAD

INDIAN INSTITUTE OF MANAGEMENT
AHMEDABAD

A Mathematical Programming Method to determine exact optimum allocation in stratified random sampling

M. RAGHAVACHARI

W.P. No. 149
February 1977

The main objective of the working paper series of the IIMA is to help faculty members to test out their research findings at the pre-publication stage

Indian Institute of Management
Ahmedabad

A Mathematical Programming Method to determine exact optimum allocation in stratified random sampling

By M. RAGHAVACHARI

Indian Institute of Management, Ahmedabad

SUMMARY

The derivation of "optimal allocation" of sample sizes to different strata in stratified random sampling does not take into account the integrality, non-negativity and upper bound restrictions on the sample sizes. One of the consequences of this is that the formula at times requires more than 100% sampling in some strata. In this paper an exact integer programming formulation is proposed and an efficient method of finding optimal solution is given. An alternative method using incremental or marginal analysis approach is also given. A numerical example illustrating the non-optimality of the current procedures and the corresponding optimal solution obtained by the methods of the paper is also presented.

Some key words : Stratified sampling; optimal allocation; Integer programming; Dynamic programming; Marginal analysis; Neyman allocation.

1. INTRODUCTION : CURRENT PROCEDURES

The well known problem of the "optimal" allocation of sa sizes to various strata in stratified random samplimg design i usually stated as :

minimize $$\text{Var}\ (\bar{y}_{st}) = \sum_{h=1}^{L} \frac{W_h^2\ S_h^2}{n_h} \left(1 - \frac{n_h}{N_h}\right) \qquad (1.$$

subject to

$$c_1\ n_1 + c_2\ n_2 + \ldots\ldots + c_L\ n_L = C \qquad \ldots \qquad (1.2)$$

where

$\bar{y}_{st}$	=	Estimate of the population mean $\bar{Y}$
L	=	Number of strata
N_h	=	Size of the stratum h in the population, h=1, ... L
S_h^2	=	Variance in the stratum h, h=1, ... L
W_h	=	N_h / N , h=1, ... L
n_h	=	Sample size in the stratum h, h=1, ... L
c_h	=	Cost of taking single observation in stratum h, h=1, ... L
N	=	$N_1 + N_2 \ldots + N_L$
C	=	Total Budget

The "optimal allocation" i.e. the optimal solution to (1.1) subject to (1.2) is given by

$$n_h/m = \frac{N_h \; S_h/(c_h)^{\frac{1}{2}}}{\sum_{h=1}^{L} N_h \; S_h/(c_h)^{\frac{1}{2}}} \qquad h=1,\ldots L \qquad (1.3)$$

with

$$m = C \; \frac{\Sigma N_h \; S_h/(c_h)^{\frac{1}{2}}}{\Sigma N_h \; S_h \; (c_h)^{\frac{1}{2}}} \qquad \ldots \qquad (1.4)$$

The current procedures is to solve for n_h from (1.3) and (1.4) and round them off to nearest integers. The rounded off solution may exceed the budget. It has also been reported that in some practical applications, the values of n_h so obtained exceed the respective stratum population sizes N_h for some h in which case more than 100% sampling from that stratum is called for by the optimal allocation formula (1.3). See, for example, Cochran (1963 p. 103). If any stratum requires more than 100% sampling, the current procedure is to allocate 100% sampling i.e. to take $n_h = N_h$ for that stratum and allocate again in the remaining strata with the budget correspondingly reduced. This procedure is repeated until one gets an allocation satisfying the upper bound restrictions on n_h. We shall see later through an example that this sequential procedure need not lead to an optimal allocation.

The purpose of this paper is to give a precise mathematical programming formulation of the problem and give exact methods of solution which gives direct integer solutions of n_h and maintain the upper bound restrictions: $n_h \leq N_h$. Two methods of solution are given in the paper. One of them is based on the dynamic programming principle and the other on the incremental or marginal analysis approach.

2. INTEGER PROGRAMMING SOLUTION

The formulation given by (1.1) and (1.2) is not complete. Integer restrictions on n_h and upper bound restrictions $n_h \leq N_h$ are not imposed. Thus n_h obtained by (1.3) and rounded off to integers may not be optimal. In this section we give the precise integer programming formulation of the problem.

Since $\sum_{h=1}^{L} W_h^2 S_h^2 / N_h$ is a given constant in (1.1), the problem of optimal allocation is given by :

maximize $$f(n_1, \ldots n_L) = - \sum_{h=1}^{L} \frac{N_h^2 S_h^2}{n_h}$$

subject

$$T(n) = T(n_1, \ldots n_L) = c_1 n_1 + c_2 n_2 + \ldots + c_L n_L = C$$

$$0 \leq n_h \leq N_h \qquad h=1, \ldots L$$

$$n_h \text{ is integer} \qquad h=1, \ldots L$$

This is a non-linear integer programming problem and belongs to the class of "distribution of effort" problems. See Wagner (1975, sec. 10.2 p. 333). This problem can be solved by using dynamic programming recursion. Assuming without lose of generality that the c_j's and C are integers (if they are not, we can multiply all these by their least common multiple) the dynamic programming recursion is

$$g_h(p) = \underset{n_h}{\text{maximum}} \{ (R_h(n_h) + g_{h-1}(p - n_h c_h) \} \quad h=1,\dots L$$

$$g_o(p) \equiv 0$$

where $p = 0,1, \dots. C$ and the maximization is done over only non-negative integers value of n_h satisfying $n_h \leq \min (p, N_h)$. Here $R_h(n_h) = - W_h^2 S_h^2/n_h$. The value sought is $g_L(C)$. This recursion gives the exact method to get optimal allocation.

We shall describe another method that has been proposed for distribution of effort problems. Note that in the optimal solution to (2.1), each $n_h \geq 1$. This is because the objective function takes infinite value when any n_h is zero. Further, for positive values of x, 1/x is a decreasing convex function so that $f(n_1, \dots. n_L)$ is a concave function of $n_1, \dots n_L$. for all positive values of $n_1, n_2, \dots n_L$. Using incremental marginal analysis approach B.Fox (1966) proposed a method for a special class of 'Distribution of Effort' problems. See also E. Kao (1976). For our problem in which the n_h's are bounded by N_h, we give below an adaptation of B.Fox method. Let n denote $(n_1, .. n_L)$.

Step 1 : $n^{(0)} = (1, 1, \ldots 1)$

Step 2 : $k = 1$

Step 3 : $n^{(k)} = n^{(k-1)} + e_i$ where e_i is the $i^{\underline{th}}$ unit vector and i is any index for which $\{R_h\ (n_h^{(k-1)} + 1) - R_h(n_h^{(k-1)})\}/c_h$ is a maximum where maximization is over $h=1, .. L$ such that $n_h^{(k-1)} < N_h$.

Step 4 : If $T\ (n^{(k)})\ \ C$, let $n^* = n^{(k-1)}$ and terminate; otherwise set k as $(k+1)$ and go to step 3.

For the case when the c_j's are not all equal, if $T(n^{(k)}) = C$ in step 4 of the algorithm, the solution is optimal; if $T(n^{(k)}) > C$, the solution obtained is nearly optimal. Thus when all the $c_j \equiv 1$ i.e. for the case of "Neyman allocation" , the algorithm yields an optimal solution. See Kao (1976) for details of the method and for a proof of optimality for this class of distribution of effort problems. A computer program was written to obtain the allocations using formula (1.3) and the dynamic programming algorithm. The dynamic programming algorithm was observed to be quite fast when it was tried on a number of problems. The dynamic programming method is quite efficient to get the exact optimal solution. The marginal analysis method, it is believed, will be more efficient than the dynamic programming method. It should be noted however that the marginal analysis method gives the exact optimal allocation only for the "Neyman allocation" case i.e. when all $c_j = 1$. Since the dynamic programming algorithm gave exact optimal allocation in reasonable length of computer time, it is recommended that this algorithm can be used effectively.

3. EXAMPLE

We present an example to compare the 'approximate' optimal allocation commonly used (formula (1.3)) with the exact optimal allocation obtained by the dynamic programming recursion to solve the integer program (2.1). The example is given below.

Table 3.1 : Data

Stratum	N_h	S_h	c_h
1	200	200	14
2	100	30	15
3	250	4	16
4	260	500	17
5	270	60	18
6	280	7	19
7	60	999	1
8	100	40	2
9	110	5	18
10	120	400	17

$C = 1200$

The optimal allocation formula (1.3) after rounding off to integers yields n_h, h=1, .. L. It was observed that n_7 = 69 which is more than N_7 = 60, which means over sampling in stratum 7. If we take a sample of size 60 from stratum 7 and reallocate 'optimally' among the other strata with a budget 1200-60(1) = 1140, we get an allocation which satisfies the upper bound restrictions on the population stratum size. The optimal integer solution by dynamic programming recursion was also obtained. The results are shown below.

Table 3.2 : Allocations

Stratum	Allocation using current method (formula (1.3))	Sample sizes allocation using (1.3) after taking 100% sampling in Stratum 7	Optimal integer program solution
1	12	12	13
2	1	1	1
3	1	1	1
4	36	36	35
5	4	4	4
6	1	1	1
7	69	60	56
8	3	3	3
9	1	1	1
10	13	13	13
Total Cost	1216	1207	1200

From the above table we can see that the rounded off solution (column 2 of table 3.2) is not feasible with the total cost exceeding the budget 1200. The optimal integer solution (column 4 of table 3.2) does not use 100% sampling in stratum 7. This shows that even the solution in column 2 of the table is not optimal. This example thus shows that the sequential method of using the formula (1.3) successively after taking 100% of the stratum size of the over sampled strata need not produce the optimal integer allocation.

ACKNOWLEDGEMENT

The author is thankful to Mr. C.S.R. Murthy of Physical Research Laboratory, Ahmedabad for writing the Computer Programs.

REFERENCES

COCHRAN, W. (1963) *Sampling Techniques*, Second Edition, New York : Wiley.

FOX, B. (1966) Discrete Optimization via marginal Analysis. *Management Science* **11**, 210-16

KAO, E.P.C. (1976) On incremental analysis in resource allocations. *Operational Research Quarterly* **27**, 759-63.

WAGNER, H. (1973) *Principles of Operations Research with applications to managerial decisions.* Prentice-Hall : New York.