On splitting of sizes for sampling procedure with inclusion probabilities proportional to size

Vijai Kumar Dwivedi
Department of Statistics, University of Botswana, Botswana
vijai.dwivedi@mopipi.ub.bw

Available online at: www.isca.in, www.isca.me
Received 25th October 2017, revised 3rd January 2018, accepted 10th January 2018

Abstract

The efficiency of Inclusion Probabilities Proportional to Size (πPS) sampling schemes varies from one another due to different sets of joint inclusion probabilities (i.e., πij’s). Dwivedi provided an algorithm-I which control on πij (i=1,2,...,N−1) i.e. on diagonal value of π. The other values of πij’s (j>i+1) remains more or less equal. Thus it is desirable to have control on other values of πij’s as well besides the diagonal ones. This paper provides a modified version of algorithm-I named as algorithm-II for a proper split of sizes X’s which starts with simultaneous control on πij’s. The algorithm-II basically contains a maximum of N-n stages, and in the s-th stage, complete splitting of Xs is achieved such that resulting πij’s satisfies condition φij<1 i.e. the non-negativity of variance estimates condition. It is presented that on an average the relative efficiency of proposed algorithm-II demonstrates the supremacy over probability proportional to size with replacement sampling scheme (PPSW).

Keywords: Selection probabilities, unequal probabilities, inclusion probabilities.

Introduction

Utilizing the nature of non-negativity condition of variance estimator (φ>0) approach, Dwivedi provided split of sizes with fewer number of attempts resulting a set of πij’s satisfying the non-negativity condition φij<1 as suggested by Hanurav1. Dwivedi2 suggested an algorithm named as algorithm-I which control on πij (i=1,2,...,N−1) i.e. diagonal value of π. The other values of πij’s (j>i+1) remains more or less equal. It is known that the efficiency of Inclusion Probabilities Proportional to Size (πPS) sampling schemes varies from one another due to different sets of joint inclusion probabilities (i.e. πij’s). Thus it is desirable to have control on other values of πij’s as well besides the diagonal ones. The control on πij may be exercised with the help of corresponding πiπj value. Because of condition 0 <Xij ≤ Xij+1 (i=1,2,3,...,N−1) the values of elements of matrix Φij=(πij,πij) increase horizontally from left to right and vertically downwards. Hence if the control on πij’s are exercised systematically, such that the resulting matrix (πij) also follows more or less the same trend as Φij, then the values of Φij are more likely to lie within reasonable limits. While the remaining values of πij’s (i ≤ N-n+1, .... N−1; j= i+1,...,N) could be controlled with a little effort by shifting of elements from one to other columns. Thus still a little trial and error is required and this could not be completely eliminated. The proposed algorithm (named as algorithm-II) thus provides control on πij’s satisfying condition φij<1. The approach of algorithm-II also depends on splitting of X’s but it basically differs from the approaches of Srivastava and Singh’s3 method and Dwivedi’s2 algorithm-I in the sense that here the splitting of X’s starts with simultaneous control on πij’s.

The algorithm-II basically contains a maximum of N-n stages, and in the s-th stage, complete splitting of Xs is achieved such that πij’s satisfies condition φij<1. Here Xs is defined as follows along with some other terms which are needed in further workout.

\[ U : N-n, \text{ the upper limit where the splitting process terminates.} \]
\[ X_0 : \text{ The size of i-th unit at s-th stage.} \]
\[ X_i : \text{ The split value of Xs in the i-th column in s-th stage and in s-th row (t=s+1, ...,U(s); t ≤ s; s ≥ i ).} \]

Where to satisfy the condition of minimum n units in each split, U(s) = (N-n+2)−s are the maximum number of columns which can accommodate the split of Xs. For the sake of computational ease it is assumed here that Xst takes only integer values. If Xst takes a non-integer values then the integer portion will be retained to satisfy this assumption.

Arrange Xs in ascending order and put them in the column of sizes, such that i-th unit is placed in the i-th row. The steps of algorithm-II are then as follows:

Step-1: Calculate \[ \pi_i = n X_i X^{-1} (i=1,2,...,N), \text{ and } U(s) = (N-n+2)-s \]

Where: \[ X = \sum_{i=1}^{N} X_i. \]
Select a value of $R < 1$ being the desired level of $\phi_{12}$.

Then $\phi_{12} \sqsubseteq R = mX_{11}^{\prime} \left[ \pi_1 \pi_2 (N-1) \right]^{-1}$

$$X_{11}^{\prime} = \pi_1 \pi_2 (N-1) R m^{-1}$$

i.e. where, $m = n(n-1)X^{-1}$.

Since $X_{11}^{\prime}$ should be at least unity, we redefine,

$$X_{11}^{\prime} = X_{11}^{\prime\prime} \text{ if } X_{11}^{\prime} \geq 1$$

$$= 1 \text{ if } X_{11}^{\prime} < 1.$$ Further, let $X_{11}^{\prime\prime} = X_{11}^{\prime} - X_{11}^{\prime\prime}$

Step-2: Define, $UU(1) = U(1) - 1$, being the total number of columns over which $X_{11}^{\prime\prime}$ can be distributed over a total number of columns not exceeding $X_{11}^{\prime\prime}$, we redefine

$$UU(1) = \begin{cases} UU(1) & \text{if } UU(1) \leq X_{11}^{\prime\prime} \\ X_{11}^{\prime\prime} & \text{if } UU(1) \geq X_{11}^{\prime\prime} \end{cases}$$

Step-3: Let $a(1) = UU(1) + 1$

Let $X_{12}^{\prime} + X_{13}^{\prime} = \ldots = X_{11}^{\prime}UU(1) = X_{11}^{\prime\prime} / UU(1)$

and, $X_{1,a(1)} = X_{11} - X_{12}^{\prime} - \ldots - X_{11}^{\prime}UU(1)$

$$= X_{11} - X_{11}^{\prime\prime} - [UU(1) - 1]X_{11}^{\prime\prime} / UU(1)$$

Step-4: Calculate, $\pi_{12} = mX_{11}^{\prime} (N-1)^{-1}$

and, $\phi_{12} = \pi_{12} \left( \pi_1 \pi_2 \right)^{-1}$

Step-5: Let $a(2) = UU(1) + 2$. Calculate for $i=3,4,\ldots,a(2)$

$$\pi_{1i} = \pi_{1,i-1} + mX_{1,i-1}^{\prime} (N+1-i)^{-1}$$

and, $\phi_{1i} = \pi_{1i} \left( \pi_1 \pi_i \right)^{-1}$

If $\pi_{1i} \geq 1$ then proceed to Step-7

Step-6: If $n=2$, then proceed to Step-8, otherwise define $a(3) = UU(1) + 3$; Calculate for $i=3,4,\ldots,a(3)$

$$\pi_{1i} = \pi_{1,a(2)}$$

and, $\phi_{1i} = \pi_{1i} \left( \pi_1 \pi_i \right)^{-1}$

then proceed to Step-8

Step-7: Calculate, $X_{11}^{\prime} = X_{11}^{\prime\prime} + 1$

$$X_{1,i-1}^{\prime} = 1 / X_{1,i-1}^{\prime\prime}$$

and, return to Step-4

Step-8: Up to this stage we have split $X_{11}$ completely and $X_{1i}$, $i>1$, partially along with exercising control on $\pi_{12}$, $\pi_{13}$, $\ldots$, $\pi_{1N}$. The residual stock at the 1st stage are calculated as follows. For $i=2,3,\ldots,a(2)$

$$X_{2i} = X_{1i} - \sum_{j=1}^{i-1} X_{1j}^{\prime\prime}$$

and for $n>2$ and $i= a(3), \ldots,N$

$$X_{2i} = X_{1i} - X_{11}^{\prime\prime}$$

These residual stock will provide other values of $\pi_{ij}$’s $(i>1, j=i+1,\ldots,N)$ after suitable shifting. These residual stocks will be the starting stocks for the second stage, where $X_{22}$ will be split completely and $X_{2i}$ ($i=3,4,\ldots,N$) will be partially exhausted. This splitting is is done in such a way that the resulting values of $\pi_{23}$, $\pi_{24}$, $\ldots$, $\pi_{2N}$ satisfy condition 3.7. It is to be noted here that this type of splitting $\pi_{i,i+1}$ will contribute towards the values of $\pi_{i+1,i+2}$.

Step-9: $UU(s)$ is the total number of columns over which $X_{ss}$ can be distributed. Obviously for the second stage, $s=2$. Since $X_{ss}$ can be distributed over a total number of columns, not exceeding $X_{ss}$, following Step-2, we redefine

$$UU(s) = \begin{cases} UU(s) & \text{if } UU(s) \leq X_{ss} \\ X_{ss} & \text{if } UU(s) > X_{ss} \end{cases}$$

Step-10: Let $a(4)=UU(s)$ and $a(5)=UU(s)-1$

Let $X_{s1}^{\prime} = X_{s2}^{\prime} = \ldots = X_{s,a(5)}^{\prime} = X_{ss} / UU(s)$

and, $X_{s,a(4)} = X_{ss} - X_{s1}^{\prime} - X_{s2}^{\prime} - \ldots - X_{s,a(5)}^{\prime}$

$$= X_{ss} - a(5) X_{ss} / UU(s)$$

Step-11: For the type of split considered here the diagonal elements of $(\pi_{ij})$ matrix can be obtained using the recurrence relation given below:
\[
X_{s,s+1} = \sum_{j=1}^{s-1} \pi_{js} - \sum_{j=1}^{s-2} \pi_{j,s-1} + m X_{s}^{'} (N-s)^{-1}
\]

also,
\[
\phi_{s,s+1} = \pi_{s,s+1} (\pi_{s}, \pi_{s+1})^{-1}
\]

Step-12: Let \(a(6) = UU(s) + s\). The off diagonal elements of above matrix \((\pi_{ij})\) can be obtained as follows:
\[
\pi_{i,j-1} = \pi_{s,i-1} + m X_{i,j-1}^{'}, \quad (N-i+1)^{-1}
\]
\[
\phi_{i,j} = \pi_{i,j} (\pi_{i,j})^{-1} \quad ; i=s+2, \ldots, a(6)
\]

If \(\pi_{s} \geq 1\), then proceed to Step-14

Step-13: If \(n=2\), then proceed to Step-15 otherwise define \(a(7) = UU(s) + s+1\) and calculate for \(i=a(7), \ldots, N\)

\[
\pi_{i,j} = \pi_{i,j} (\pi_{i,j})^{-1}
\]

and proceed to Step-15.

Step-14: Calculate,
\[
X_{s}^{'}, \quad X_{s,i-1}^{'}, \quad X_{s,i-1}^{'}, \quad \text{return to Step -11}
\]

Step-15: In this way complete splitting of \(X_{22}\) and partial splitting of \(X_{2i}\) \((i>2)\) have been achieved along with control on \(\pi_{23}, \pi_{24}, \ldots, \pi_{2N}\). Such that the condition 3.7 is satisfied. Now the residual stocks from the second stage are used as starting stock for 3-rd stage. These are obtained using the following formulae;
\[
X_{s+1,i} = X_{s}^{'}, \quad (i=s+1, \ldots, UU(s) + s+1)
\]

and for \(n > 2\),
\[
X_{s+1,s} = X_{s} - X_{ss}
\]

The Steps – 9 to 15 are then repeated. This procedure is continued until all the values of \(s\) are exhausted i.e. \(s=3, 4, \ldots, \text{etc.}\)

If at any stage some \(X_{s+1,i}\) obtained using the above formulae, becomes negative then the process is terminated after completing the \((s-1)\)-th stage.

Step-16: Thus this process exhausts the first \(\leq N-n\) stocks with simultaneous control on \(\pi_{i,j}\)’s. Only \(n'\) residual stocks are left which are to be adjusted in such fashion that condition of non-negativity condition is satisfied. These residual stocks can be adjusted with little effort by shifting some of the elements from one to other columns. After achieving complete shifting (i.e. final splitting) in this manner, sampling scheme Srivastava and Singh’ applied.

Example of splitting and corresponding set of \(\pi_{i,j}\)’s: To illustrate the above suggested algorithm-II, we consider a numerical example consisting of three populations A, B and C given by Yates and Grundy as follows:

Table-1: Three artificial population of size, \(N = 4\).

<table>
<thead>
<tr>
<th>Population</th>
<th>(U_1)</th>
<th>(U_1)</th>
<th>(U_2)</th>
<th>(U_3)</th>
<th>(U_4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size</td>
<td>(X_i)</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Population A</td>
<td>(Y_i)</td>
<td>0.5</td>
<td>1.2</td>
<td>2.1</td>
<td>3.2</td>
</tr>
<tr>
<td>Population B</td>
<td>(Y_i)</td>
<td>0.8</td>
<td>1.4</td>
<td>1.8</td>
<td>2.0</td>
</tr>
<tr>
<td>Population C</td>
<td>(Y_i)</td>
<td>0.2</td>
<td>0.6</td>
<td>0.9</td>
<td>0.8</td>
</tr>
</tbody>
</table>

The scale of \(X_i\)’s has been changed by a constant multiplier 10. After arranging in the ascending order the values are 10, 20, 30 and 40 respectively. Here \(N=4, n=2, U=2, U(1)=3\) and \(U(2)=2\). The process terminates after 2\textsuperscript{nd} stage. The \(\pi_{1}, \pi_{2}, \pi_{3}\) and \(\pi_{4}\) values are 0.2, 0.4, 0.6 and 0.8 respectively. The value of \(R\) is chosen as 0.5. Following Step-1, \(X_{11} = 7\) and \(X_{11} = 3\). Also Step-2 gives \(UU(1) = 2\). Then following Step-3, \(X_{12} = 1\) and \(X_{13} = 2\). Then Steps-4 and 5 provide \(\pi_{12}=0.047, \pi_{13}=0.057\) and \(\pi_{14}=0.05\). Residual stock from the first stage, being the starting stocks for the second stage are then given by \(X_{22} = 13, X_{23}=22\) and \(X_{24}=30\) (Step-9). Similarly split of \(X_{22}\) is obtained as \(X_{21} = 6\) and \(X_{22} = 7\) and the starting stock for 3\textsuperscript{rd} stage are \(X_{31}=16\) and \(X_{34}=17\) (Steps-9 to 15). Since \(U(3) = 0\), the process terminates. Here \(X_{31}\)’s are residual stocks which are adjusted with a little effort in such a manner that condition suggested by Srivastava and Singh’ is satisfied. The final split achieved is depicted below in Table-2.

Table-2: Final split of sizes with at least 2 nonzero elements in each column.

<table>
<thead>
<tr>
<th>Population units</th>
<th>Sizes ((X_i))'s</th>
<th>Columns (Groups)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>7</td>
</tr>
<tr>
<td>Total</td>
<td>100</td>
<td>28</td>
</tr>
</tbody>
</table>
Using the formulae given by Srivastava and Singh the corresponding $\Phi$ matrix for split in Table-2 thus obtained and is given below:

$$\Phi = \begin{bmatrix} 0.58 & 0.47 & 0.60 \\ 0.40 & 0.80 \\ 0.93 \end{bmatrix}$$ (1)

Following the initial split obtained using the method of Srivastava and Singh the corresponding $\Phi$ thus obtained is given below

$$\Phi = \begin{bmatrix} 0.83 & 0.55 & 0.42 \\ 0.28 & 0.83 \\ 0.97 \end{bmatrix}$$ (2)

It is clear from matrix (1) that algorithm-II provides a set of $\pi_i$’s for which $\phi_i$’s lie within a closer limit than that obtained using the method of Srivastava and Singh as given in matrix (2).

The above numerical example having small value of N and n is chosen to illustrate the various steps involved in algorithm-II. The method Srivastava and Singh provided a split in which the range of $\phi$ is between 0.28 and 0.97 whereas the proposed algorithm-II provided a split for which these values lie between 0.43 and 0.93 which is more desirable from the efficiency point of view. Thus algorithm-II is clearly preferable.

**Relative efficiency**

For the population depicted in Table-1, the exact variance of Horvitz Thompson (HT) estimator of the population total $Y$ resulting from algorithm-II is compared with probability proportional to size with replacement (PPS WR) sampling scheme and algorithm-I is presented in Table-3.

**Conclusion**

On an average the relative efficiency of both the algorithms shows the supremacy over PPSWR. Algorithm-I is more subjective than algorithm-II and also more sensitive to the population characteristics. The algorithm-II demonstrated better for population C (Table-3).

**Acknowledgement**

Author is thankful and indebted to Prof. A.K. Srivastava, Former Principal Scientist, Indian Agricultural Statistics Research Institute (IASRI), New Delhi for his valued supervision to carry out this research work.

<table>
<thead>
<tr>
<th>Population</th>
<th>PPS WR</th>
<th>Algorithm-I R-Values</th>
<th>Algorithm-II R-Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.500</td>
<td>0.283 0.312 0.333</td>
<td>0.292 0.322 0.338</td>
</tr>
<tr>
<td>B</td>
<td>0.500</td>
<td>0.283 0.312 0.333</td>
<td>0.292 0.322 0.338</td>
</tr>
<tr>
<td>C</td>
<td>0.125</td>
<td>0.067 0.053 0.047</td>
<td>0.057 0.048 0.042</td>
</tr>
<tr>
<td>Average</td>
<td>0.375</td>
<td>0.188 0.225 0.238</td>
<td>0.213 0.231 0.239</td>
</tr>
<tr>
<td>Relative Efficiency</td>
<td>100</td>
<td>198 166 157</td>
<td>175 162 156</td>
</tr>
</tbody>
</table>
References


