LINEAR PROGRAMMING METHODOLOGY USED FOR DISCLOSURE AVOIDANCE PURPOSES AT THE CENSUS BUREAU

Laura Voshell Zayatz *
Bureau of the Census

KEY WORDS: Tabular Data, Operations Research, Confidentiality

I. Introduction to the Problem

The Bureau of the Census is responsible for collecting information about the country's business establishments under a pledge of confidentiality and for publicly releasing this information without disclosing individual responses. The Bureau publishes the information in the form of two- or three-dimensional **additive, non-negative** tables.

There are sometimes cell values in the tables that the Bureau cannot publish without risking a violation of the confidentiality pledge. For example, if there is only one firm contributing to a cell, the Bureau cannot publish that cell's value. The actual formula used for deciding which table cells cannot be published is confidential, however, in general, cell values that are highly dominated by a small number of respondents are considered to possess a high risk of disclosure. The Bureau's current practice is to suppress any cell where **n** or fewer respondents make up **k** or more percent of that cell's value. The values of n and k are confidential. Any cell values that violate this criterion are called primary suppressions.

Because the tables that the Bureau publishes are additive, it is usually not enough to suppress only those cell values that violate the **n-k rule**. An outsider could obtain the suppressed values through addition and subtraction. Therefore, the Bureau must suppress other cell values in the tables. The other values that are chosen for suppression for this reason are called complimentary suppressions. Although network methodology handles the problem of choosing complementary suppressions quite nicely for two-dimensional tables, linear programming (LP) methodology has some advantages for three-dimensional tables. This paper describes the technique of using linear programming to find complimentary suppression patterns for three-dimensional tables.

II. Minimizing the Total Value Suppressed While Providing Sufficient Protection

The Bureau's goal is to publish as much valuable information as possible without violating the confidentiality pledge. Thus the Bureau attempts to choose complimentary suppressions in such a way that the sum of the values chosen for complimentary suppression is minimized while still ensuring that the suppressions are large enough so that the individual responses in primary suppressions are protected.

Consider the two-dimensional additive table below.

100	12	5	250	367
12	12	5	5	34
40	200	90	300	630
5	70	50	5	130
157	294	150	560	1161

Suppose that the cell in the first row and first column is a primary suppression. We identify it as such in the table below.

P	12	5	250	367
12	12	5	5	34
40	200	90	300	630
5	70	50	5	130
157	294	150	560	1161

If the table above were published, an outsider could determine the exact value of the primary suppression by subtraction.

$$P = 367 - 12 - 5 - 250 = 100$$

Suppose we add some complimentary suppressions to the table as seen below.

P	12	C_{13}	250	367
12	12	C_{23}	C_{24}	34
40	200	90	300	630
C_{41}	70	50	C_{44}	130
157	294	150	560	1161

Using some simple algebra, an outsider could now estimate that the primary suppression value was between 95 and 105. (From Column 3 we know that $0 <= C_{13} <= 10$. Using this information and the non-negativity constraint, Row 1 implies that $95 <= P <= 105$). This may or may not be enough protection for this primary suppression. Whenever a cell has been designated as a primary suppression, the Bureau calculates a value, **prot**, such that if an outsider can use algebra to say at best that P could lie anywhere in the interval between P - prot and P + prot then the individual responses contained in that primary suppression are considered sufficiently protected. As stated before, the values of n and k are confidential. The method of calculating prot is also confidential. Say that for this primary suppression, prot = 15. Thus, we need to add more complimentary suppressions, as in the table below.

P	**C**	**C**	250	367
C	**C**	**C**	**C**	34
40	200	90	300	630
C	70	50	**C**	130
157	294	150	560	1161

The best that an outsider can do with algebra or linear programming techniques is to estimate that the primary suppression value was between 83 and 117. This amount of protection would now be considered sufficient since $83 <= 100 - 15$ and $117 >= 100 + 15$.

To ensure that our primary suppression in the example above was sufficiently protected, we had to suppress a total cell value of $5 + 5 + 5 + 5 + 5 + 12 + 12 + 12 = 61$. Note that we could have chosen a different set of complimentary suppressions as shown below.

P	12	5	**C**	367
12	12	5	5	34
C	200	90	**C**	630
5	70	50	5	130
157	294	150	560	1161

This pattern provides the necessary protection, is simpler, and suppresses fewer table cells. But the total value of our complimentary suppressions (which is what we are attempting to minimize) is $250 + 40 + 300 = 590$.

The example above shows possible complimentary suppression patterns for a table with one primary suppression. Many of the Bureau's tables have several primary suppressions. If that is the case, the current practice is to choose complimentary suppressions for one primary suppression at a time. We call this processing one primary suppression at a time. Each time we process a primary suppression, we suppress all cell values in the table that are chosen as complements for that primary. As one could imagine, large tables with many primary suppressions have very complicated complimentary suppression patterns.

III. Linear Programming Methodology versus Network Flow Methodology

Linear programming methodology can be used to find complimentary suppression patterns in two- and three-dimensional tables (Lougee-Heimer 1989). Network flow methodology may also be used to find complimentary suppression patterns, but in two-dimensional tables only (Cox 1987; Cox 1980; Cox, Fagan, Greenberg, and Hemmig 1986; Sullivan and Zayatz 1991; Rowe 1991). In fact, algorithms based on network flow methodology work faster than those based on linear programming methodology for this application. Since the algorithms yield the same suppression patterns, we recommend using network flow methodology for finding complimentary suppression patterns in two-dimensional tables.

Unlike linear programming methodology, network methodology cannot consider all of the additive relationships in a three-dimensional table simultaneously. One could use network methodology to examine every two-dimensional table contained within a three-dimensional table and apply complimentary suppressions to each one independently, reprocessing some of those two-dimensional tables as necessary. In other words, for each level, complimentary suppressions are applied to the two-dimensional table containing the cell values in that level for all rows and columns. Then, for each row, complimentary suppressions are applied to the two-dimensional table containing the cell values in that row for all columns and levels. Then, for each column, complimentary suppressions are applied to the two-dimensional table containing the cell values in that column for all levels and rows. After processing each of these two-dimensional tables, it may be necessary to reexamine some of them and possibly apply more complimentary suppressions. This is because when

processing a certain two-dimensional table, we may apply a complimentary suppression to a cell contained in another two-dimensional table(s) that was previously processed. If this happens, we must reexamine the previously processed table(s) to see if more complimentary suppressions are necessary. When all of the two-dimensional tables have been processed, and reprocessed if necessary, we are finished.

As tedious a task as this may seem, it is usually still faster than using linear programming methodology. For tables with many cells and many primary suppressions, the difference in time is substantial. As one might guess, the network method also results in more cells being chosen for complimentary suppression, but adjustments to the cost function can compensate for this problem. A more serious problem with this repetitive network flow approach is that even after applying the technique, some primary suppressions may remain unprotected. Consider the three-dimensional table below.

Level 1

	C1	C2	C3	C4	Total
R1	P	P	3	4	10
R2	P	P	7	8	26
R3	9	10	11	12	42
R4	P	P	P	P	58
R5	P	P	P	P	74
Total	45	50	55	60	210

Level 2

	C1	C2	C3	C4	Total
R1	P	P	23	24	90
R2	P	P	27	28	106
R3	29	30	31	32	122
R4	P	P	P	P	138
R5	P	P	P	P	154
Total	145	150	155	160	610

Level 3

	C1	C2	C3	C4	Total
R1	P	P	3	4	10
R2	P	P	7	8	26
R3	9	10	11	12	42
R4	13	14	P	P	58
R5	17	18	P	P	74
Total	45	50	55	60	210

Level 4

	C1	C2	C3	C4	Total
R1	P	P	P	P	90
R2	P	P	P	P	106
R3	29	30	31	32	122
R4	33	34	P	P	138
R5	P	38	P	P	154
Total	145	150	155	160	610

Total Level

	C1	C2	C3	C4	Total
R1	44	48	52	56	200
R2	60	64	68	72	264
R3	76	80	84	88	328
R4	92	96	100	104	392
R5	108	112	116	120	456
Total	380	400	420	440	1640

If we separately examine every two-dimensional table contained within this three-dimensional table, every primary suppression appears to be protected with a prot value of at least 1. However, using linear programming techniques which look at the three-dimensional table as a whole, one can conclude that the value of the primary suppression in row 5, column 1, level 4 is exactly 37.

For three-dimensional tables, we recommend using the linear programming formulation described in Section IV for finding complimentary suppression patterns. However, if time constraints render this method unusable, we recommend using network flow methodology repetitively as described above for finding complimentary suppression patterns. A much smaller and faster linear program may then be used to locate any primary suppressions that are not sufficiently protected. This smaller linear program is described in Section V. More complimentary suppressions may then be added to protect those primary suppressions.

IV. Cell Suppression LP Formulation for Three-Dimensional Tables

The model that can be used to find complimentary suppressions for a primary suppression in row r, column c, level l in a three-dimensional additive m x n x p table is as follows:

Variables:

D_{ijk1} and D_{ijk2}, for all i = 1,...,m, j = 1,...,n, k = 1,...,p except when (i=r **and** j=c **and** k=l)

Constraints:

$$\sum_{i=1}^{m} (D_{ijk1}-D_{ijk2}) = 0 \text{ for all } j = 1,...,n,\ k = 1,...,p$$

$$\sum_{j=1}^{n} (D_{ijk1}-D_{ijk2}) = 0 \text{ for all } i = 1,...,m,\ k = 1,...,p$$

$$\sum_{k=1}^{p} (D_{ijk1}-D_{ijk2}) = 0 \text{ for all } i = 1,...,m,\ j = 1,...,n$$

D_{ijk1} <= cell value in row i, column j, level k for all i = 1,...,m, j = 1,...,n, k = 1,...,p except when (i=r **and** j=c **and** k=l)

D_{ijk2} <= cell value in row i, column j, level k for all i = 1,...,m, j = 1,...,n, k = 1,...,p except when (i=r **and** j=c **and** k=l)

D_{rcl1} = value of prot such that if an outsider can use algebra to say at best that the value of P could lie anywhere in the interval between P - prot and P + prot then the individual responses contained in the primary suppression are sufficiently protected

$D_{rcl2} = 0$

Objective Function:

$$\text{Min} \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{p} (D_{ijk1} + D_{ijk2}) * \text{cost of suppressing the cell value in row i, column j, level k}$$

where the cost of suppressing the cell value in row i, column j, level k is calculated according to the following function.

i) 0 if the value is a primary suppression or if the value was suppressed as a complement when another primary suppression was previously processed

ii) 999999999 (a very large positive number) if the cell value is zero (the Bureau does not want to suppress any zero valued cells)

iii) the actual cell value for all other cases

Results:

If either D_{ijk1} or D_{ijk2} is greater than 0, the cell in row i, column j, level k is suppressed for all i = 1,... ,m, j = 1,...,n, and k = 1,...,p.

This linear programming formulation can be used to find complimentary suppression patterns in tables with one or more primary suppressions. It **does not yield optimal solutions** in that it does not give a set of complimentary suppressions with minimum total value that sufficiently protect all primary suppressions in a table. To obtain an optimal solution, one could use integer programming methodology (Sullivan and Rowe 1992), however the use of such methodology is computationally impractical. In fact, the cell suppression problem was shown to be NP-hard (Kelly 1990). This means that the cell suppression problem is in a class of problems for which there is no known computationally practical method for obtaining optimal solutions and little hope for finding such a method. Linear programming methods, however, do offer good solutions that provide sufficient protection. Three ways of improving the results are discussed in (Zayatz 1992).

V. Location of Under-suppression LP Formulation for Three-Dimensional Tables

As stated before, if time constraints render the

linear programming method unusable, we recommend using network flow methodology repetitively as described above for finding complimentary suppression patterns. A much smaller and faster linear program may then be used to locate any primary suppressions that are not sufficiently protected. The model that can be used to test whether or not a primary suppression in row r, column c, level l is sufficiently protected in a three-dimensional additive m x n x p table is as follows:

Variables:

S_{ijk} for all i, j, and k such that the cell in row i, column j, level k is either a primary or a complimentary suppression

Constraints:

There are n * p additive row relationships, m * p additive column relationships, and n * m additive level relationships in a three-dimensional table. For each additive relationship that contains at least one suppressed value, we have one of the following two constraints.

1. If the total is not suppressed, then the sum of the suppressed values (the S_{ijk}'s) in that relationship must equal the total value minus the sum of the other published values.

2. If the total is suppressed, then the total value minus the other suppressed values must equal the sum of the published interior values.

Objective Function:

When attempting to find an upper bound for a suppressed cell in row i, column j, level k, the objective function is Max S_{ijk}.

When attempting to find a lower bound for a suppressed cell in row i, column j, level k, the objective function is Min S_{ijk}.

Results:

In each of the cases described directly above, the resulting value of the objective function is the desired bound.

VI. Conclusions and Recommendations

As stated previously, for three-dimensional tables, we recommend using the linear programming formulation described in Section IV for finding complimentary suppression patterns. However, if time constraints render this method unusable, we recommend using network flow methodology repetitively for finding complimentary suppression patterns. The linear program described in Section V may then be used to locate any primary suppressions that are not sufficiently protected. More complimentary suppressions may then be added by hand or by some other technique to protect those primary suppressions.

VII. References

Cox, Lawrence H. (1987), "New Results in Disclosure Avoidance for Tabulations," International Statistical Institute: Proceedings of the 46th Session, pp. 83-84.

Cox, Lawrence H. (1980), "Suppression Methodology and Statistical Disclosure Control," Journal of the American Statistical Association, Volume 75, Number 370, Theory and Methods Section, American Statistical Association, Washington, D.C.

Cox, Lawrence H., Fagan, James T., Greenberg, Brian V., and Hemmig, Robert (1986), "Research at the Census Bureau into Disclosure Avoidance Techniques for Tabular Data," Proceedings of the Section on Survey Research Methods, American Statistical Association, Washington, D.C., pp 388-393.

Kelly, James P. (1990), "Confidentiality Protection in Two- and Three-Dimensional Tables," Ph.D. Dissertation, University of Maryland, College Park, Maryland.

Kelly, James P., Golden, Bruce L., and Assad, Arjang A. (1990), "Cell Suppression: Disclosure Protection for Sensitive Tabular Data," Working Paper Series MS/S 90-001, University of Maryland, College Park, Maryland.

Lougee-Heimer, Robin (1989), "Guarantying Confidentiality: The Protection of Tabular Data," Master's Thesis, Department of Mathematical Sciences, Clemson University.

Rowe, Errol (1991), "Some Considerations in the Use of Linear Networks to Suppress Tabular Data," Proceedings of the Section on Survey Research Methods, American Statistical Association, Washington, D.C., pp. 357-362.

Sullivan, Colleen and Zayatz, Laura (1991), "A Network Flow Disclosure Avoidance System Applied to the Census of Agriculture," Proceedings of the Section on Survey Research Methods, American Statistical Association, Washington, D.C., pp. 363-368.

Sullivan, Colleen and Rowe, Errol (1992), "A Data Structure and Linear Programming Technique to Facilitate Cell Suppression Strategies," Proceedings of the Section on Survey Research Methods, American Statistical Association, Washington, D.C., to appear.

Zayatz, Laura (1992), "Using Linear Programming Methodology for Disclosure Avoidance Purposes," to be presented at the Seminar on Statistical Confidentiality, Dublin, Ireland.

* This paper reports the general results of research undertaken by Census Bureau staff. The views expressed are attributable to the author and do not necessarily reflect those of the Census Bureau.