International Journal of Electronic Commerce Studies
Vol.6, No.2, pp.233-242, 2015
doi: 10.7903/ijecs.1247

MODELING DATA INTEGRITY UNDER STOCHASTIC LINEAR CONSTRAINTS

Lee-Pin Shing
Virginia Tech
926 Kabrich St, Blacksburg, VA 24060, USA
shingle@vt.edu

Lee-Hur Shing
Virginia Tech
1204 Highland Circle, Blacksburg, VA 24060, USA
leehurshing@yahoo.com

Marn-Ling Shing
University of Taipei
1 Ai-Kuo West Road, Taipei, Taiwan, R.O.C.
shing@tmue.edu.tw

Chen-Chi Shing
Radford University
Box 6933, Radford University, Radford, VA 24142, USA
cshing@radford.edu

ABSTRACT

The most commonly used data integrity models today are Bibba, Wilson-Clark and Chinese models. These models are designed for both data integrity protection and confidentiality. Many optimization problems are related to linear programming. In practice, these variables involved are probabilistic. This paper proposes a data integrity model based on data anomalies assuming data are under stochastic linear constraints. An algorithm is constructed using the simplex method to find confidence intervals for the problem solutions. In the end the results from Monte Carlo simulation are compared with those from simplex method.

Keywords: Data Integrity Model, Stochastic Linear Programming, Risk Management Model

1. INTRODUCTION

In December 2013, forty million credit/debit card holders who had shopped at Target had their accounts breached. It took Target nearly one month after November 27 to notice[1].With multiple ways to hack into a data system, it has become more difficult than ever to design better Internet security features to fortify systems with larger data. One of the most common ways to detect breaches is to find anomalies in the data. Developing an algorithm to detect these anomalies is based on data integrity. Anomalies can be found more easily with better way to distinguish data.

Data integrity guarantees that data is accurate and consistent during its transmission, process and storage. Bibba proposed the first data integrity model[2]. The Bibba model incorporates security levels to deal with untrustworthy subjects, which are used in Bell-Lapadula's confidentiality modeling[3]. This imposes too many restrictions on data processing and storage. Based on business transactions, Clark-Wilson's model[2] uses constrained data items, transformation and integrity verification procedures, well-formed transactions, and separation of duty to achieve data integrity. To avoid conflicts of interest by building a business data set with different security levels, the Chinese Wall model[2] was proposed in 1989 for both confidentiality and data integrity. This paper discusses data integrity only under stochastic linear constraints.

A linear programming problem is common for modeling a simple business problem[4]. The simplex method is customarily used to solve the problem. For example, to check the validity for the balance of a personal bank account, the total amount D of deposit minus a total amount W of withdrawal must be equal to the balance B. This can be represented by the linear constraint D - W = B. In risk management we often need to optimize a linear expression using certain variables subject to a set of constraints[5]. These m are subject to errors. A natural way to model data integrity under linear constraints is to model the problem stochastically, using either the same continuous or discrete distributions[6]. The next sections introduce a linear programming problem and then solve it with the simplex method. A natural extension of the technique is used when all variables involved are random variables. Finally, a Monte Carlo method[7] is used to calculate the probability in a confidence interval for the objective function.

2. LITERATURE REVIEW

This section introduces the definition of a linear problem, and then uses a standard form to solve it[6].

Definition 1. A standardized maximization linear programming (smaxlpp) is to find the maximum value of a linear objective function $f(x_1, \dots, x_n) = c_1x_1 + \cdots + c_nx_n$, where $x_i \geq 0$, $i = 1$,…,n and constraints

$$a_{11}x_1 + \cdots + a_{1n}x_n \leq b_1$$

…

$$a_{m1}x_1 + \cdots + a_{mn}x_n \leq b_m$$

Example 1. Maximize u= x + 2y given (x ≥ 0, y≥ 0) subject to the constraint 2x + 3y ≤ 12, and –x + y ≤ 1. This is a smaxlpp.

Definition 2. A standardized minimization linear programming (sminlpp) is to find the minimum value of a linear objective function $f(x_1, \dots, x_n) = c_1x_1 + \cdots + c_nx_n$, where $x_i \geq 0$, $i = 1$,…,n and constraints

$$a_{11}x_1 + \cdots + a_{1n}x_n \geq b_1$$

…

$$a_{m1}x_1 + \cdots + a_{mn}x_n \geq b_m$$

Example 2. Example 1 is equivalent to the problem: Minimize –u = -x - 2y given (x ≥ 0, y≥ 0) subject to the constraint 2x + 3y ≤ 12, and –x + y ≤ 1. Or (let X = -x, Y = -y, Z = -z): Minimize v = -u = X + 2Y subject to the constraint –2X-3Y ≤ 12, and X-Y ≤ 1. Or (multiply -1 on both sides of constraints) Minimize v = -u = X + 2Y subject to the constraint 2X + 3Y≥ -12, and –X + Y≥ -1. This is a sminlpp. Note: Example 2 shows that any sminlpp can be changed to a smaxlpp. Therefore we can consider only the smaxlpp form in any lpp. Based on the duality theorem, they have the same solution.

Definition 3. A canonical lpp is a smaxlpp in which all constraints are represented by equality instead of inequality using slack variables s_i, $i = 1$,…,m. That is, is to find the maximum value of a linear objective function $f(x_1, \dots, x_n) = c_1x_1 + \cdots + c_nx_n$, where $x_i \geq 0$, $i = 1$,…,n and

$$a_{11}x_1 + \cdots + a_{1n}x_n + s_1 = b_1$$

…

$$a_{m1}x_1 + \cdots + a_{mn}x_n + s_m = b_m$$

The distinguishable variables s_1, …, s_m in the constraints are called basic variables. And x_1, .., x_n are called non-basic variables. The basic point is where non-basic variables are set to 0 and solve for basic variables. The feasible basic point is the basic point where all coordinates are non-negative.

Definition 4. The canonical lpp is perfect if the basic point is feasible and the objective function depends only on non-basic variables (if not, convert basic variables into non-basic variables using constraints).

Example 3. The canonical lpp for Example 1 is to maximize u = x + 2y given (x ≥ 0, y ≥ 0) subject to the constraint 2x + 3y + r = 12, and –x + y + s = 1. The basic variables are r and s. The non-basic variables are x and y. The basic point is (x, y, r, s) = (0, 0, 0, 12, 1) and is feasible. Therefore the canonical lpp is perfect.

Definition 5. A standard canonical lpp is a canonical lpp in which the negative of the basic variable is on the right side of each constraint.

Example 4. The standard canonical lpp for Example 1 is to maximize u= x + 2y given (x ≥ 0, y ≥ 0) subject to the constraint 2x+3y -12 = -r, and –x + y - 1 = -s.

The technique for solving a standard canonical lpp is the simplex method, as described in the next section.

3. METHODS

The simplex will work on a simplex tableau in Definition 6[5].

Definition 6. The simplex tableau consists of coefficient entries with columns (non-basic variables and constants 1) and rows (constraints = negative basic variables, and objective function). For a standard canonical lpp with objective function $u = c_1x_1 + \cdots + c_nx_n$, where $x_i \geq 0,\ i = 1,\ldots,n$ and

$$a_{11}x_1 + \cdots + a_{1n}x_n - b_1 = -x_{n+1}$$

$$\ldots$$

$$a_{m1}x_1 + \cdots + a_{mn}x_n - b_m = -x_{n+m}$$

$$c_1x_1 + \cdots + c_nx_n + d * 1 = u, \text{ where } d = 0 \text{ and}$$

where $x_1, .., x_n$ are non-basic variables and $x_{n+1}, .., x_{n+m}$ are basic variables, the simplex tableau is constructed using all coefficients involved as shown in Table 1.

Example 5. The simplex tableau for Example 4 is in Table 2.

In order to get a perfect canonical lpp, all columns except the last one in the last row of the simplex tableau must be negative. To achieve this, we must follow a pivot rule, described below.

Table 1. Simplex tableau

x_1	…	x_n	1	=
a_{11}	…	a_{1n}	$-b_1$	$-x_{n+1}$
a_{21}	…	a_{2n}	$-b_2$	$-x_{n+2}$
…	…	…	…	…
a_{m1}	…	a_{mn}	$-b_m$	$-x_{n+m}$
c_1	…	c_n	d	u

Note: The last column in the table is used for explanation only.

Table 2. Simplex tableau for Example 4

x	y	1
2	3	- 12
-1	1	- 1
1	2	0

Definition 7. Pivot a_{ij}: change the basic variable in the ith row of constraints to a non-basic variable and change the jth column of non-basic variables into basic variables.

Pivot row i: the row in the simplex tableau corresponding to the ith constraint equation Before pivoting ith row: $a_{i1}x_1 + \cdots + a_{in}x_n - b_i = -x_{n+i}$. After pivoting (where i is the pivot row index and j is the pivot column index):

(1) The new kth constraint equation (k=i) becomes
$(a_{i1}/a_{ij})x_1+\ldots+(1/a_{ij})\,x_{n+i}+\ldots+(a_{in}/a_{ij})\,x_n-(b_i/a_{ij})=-x_j$

(2) kth constraint ($k\neq i$): $a^*_{k1}\,x_1+\ldots+cx_{n+i}+\ldots+a^*_{kn}x_n-b^*_k=-x_{n+k}$
where ($p\neq j$)

$a^*_{kp}=a_{kp}-(a_{kj}\,a_{ip}/a_{ij})$, $c=-\ a_{kj}/a_{ij}$ (when p=j),

$b^*_k=(b_i\,a_{kj}/a_{ij})-b_k$

(3) new objective function: $c^*_1\,x_1+\ldots+Dx_{n+i}+\ldots+c^*_nx_n+d^*=u$
where ($q\neq j$)

$c^*_q=c_q-(c_j\,a_{iq}/a_{ij})$, $D=-\ c_j/a_{ij}$ (when q=j)

$d^*=d+(c_j\,b_i/a_{ij})$

The simplex method used to solve the standard lpp is described below[5]:

Simplex Method Algorithm for Solving a Perfect Standard Canonical lpp

Goal: We want to get all C_i negative which are the entries of the last row, coefficients of the objective function, in simplex tableau

1. Choose some positive C_j

   Note: If all the entries a_{ij} in the column above C_j are negative or 0, then lpp has no solution.

2. Pivot at positive a_{ij} for which b_i/a_{ij} is min

   Note: The result is a perfect standard canonical lpp, i.e. the numbers in the constants column in the simplex tableau are either negative or zero. The value of the objective function is increasing if $b_i > 0$, unchanged if $b_i = 0$.

3. Repeat steps 1 and 2.

   Note: there are three possibilities:

   1) Reach all C_i negative- find the solution

   2) Reach all the entries a_{ij} in the column above C_j are negative or 0, that is, lpp has no solution

   3) Indefinite looping (not discussed)

Following the simplex algorithm, for Example 5, the results will be shown in Example 6.

Example 6. Use simplex method to find the optimal solution: Pivot column Index =1, min =4.000000. Pivot Row Index =1

5.000000 -3.000000 -9.000000

-1.000000 1.000000 -1.000000

3.000000 -2.000000 2.000000

****

Use Simplex Method to find the optimal solution: Pivot column Index =0, min =1.800000. Pivot Row Index =0

0.200000 -0.600000 -1.800000

0.200000 0.400000 -2.800000

-0.600000 -0.200000 7.400000

****

The maximum is 7.40. The optimal value occurs at (1.80, 2.80), optimal x = 1.80, optimal y = 2.80.

STOCHASTIC LINEAR PROGRAMMING

In this section all variables discussed in the previous sections are random variables. A linear programming problem can be extended to a stochastic linear programming problem using random variables. The problem involves calculating probability within the n-dimensional spaces formed by all constraints along with optimizing the objective function. In order to simplify the problem, we can assume that all random variables are independently distributed (or iid) as a normal distribution with mean m_1 and standard deviation g_1^2 or $N(m_1 , g_1^2)$. A simple example of calculating the probability can be shown in Example 7.

Example 7. Maximize u= X + 2Y given ($X \geq 0$, $Y \geq 0$) subject to the constraint $2X + 3Y \leq 12$, and $-X + Y \leq 1$, where X and Y are iid $N(m_1 , g_1^2)$. Suppose that f(x) and f(y) is the probability density function of $N(m_1 , g_1^2)$. Under the constraint: $2x + 3y \leq 12$,

$$\int_0^{\infty} \int_0^{4-\frac{2}{3}x} f(y)dyf(x)dx \qquad (1)$$

And under the constraint: $-x + y \leq 1$,

$$\int_0^{\infty} \int_0^{1+x} f(y)dyf(x)dx \qquad (2)$$

We can use Gaussian error function erf to calculate the integrations.

Definition 8. A Gaussian error function erf is defined as $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-\frac{t*t}{2}) dt$, where exp is the exponential function.

Theorem 1. If X distributes as a $N(m_1 , g_1^2)$, then

$$\int_0^x f(x)dx = \frac{1}{2} \text{erf} (\frac{x - m_1}{g_1\sqrt{2}}) \qquad (1)$$

The integration in Example 7 can be expressed in Example 8.

Example 8. The results for both (1) and (2) are

$$(1) = \frac{1}{2}\int_0^{\infty} dx \operatorname{erf}\left(\frac{4 - 2/3\, x - m_1}{g_1\sqrt{2}}\right), \text{ and } (2) = \frac{1}{2}\int_0^{\infty} dx \operatorname{erf}\left(\frac{1 + x - m_1}{g_1\sqrt{2}}\right).$$

In the next section, a Monte Carlo method can be used to approximate the integration under the stochastic lpp like in Example 7. Because most of the values of the random variables occur around averages, they can be calculated using the simplex method on the corresponding standard canonical lpp. Finally, we can construct a confidence interval for the objective function.

5. RESULTS

Monte Carlo integration uses computers to generate random numbers in order to approximate the calculation of integrations. It is used when the integration is too difficult to solve, as with processes in nuclear reactions. In this section, we attempt to solve the problem in Example 7 using Monte Carlo integration[6]. The following algorithm is proposed for calculating the confidence interval around the expected optimal value, using means of the random variables involved, assuming X and Y are independent and X distributes as $N(m_1, g_1^2)$ and Y distributes as $N(m_2, g_1^2)$.

Monte Carlo Method Algorithm for Finding Confidence Interval on Example 1:

Case 1. If m_1, m_2, and g_1 are known.

(1) Loop N times for the following steps:

Step 1: Generate standard normal N(0, 1) random variates x1 and y1 using a Accept-Reject Method[4] by randomly generate 2 numbers r1 and r2 between 0 and 1 first and then

Loop when s > = 1

temp1 = 2.0 * r1, temp2 = 2.0*r2, $s = temp1^2 + temp2^2$

$x1 = temp1 * \sqrt{-2\frac{\log(s)}{s}}$, $y1 = temp2 * \sqrt{-2\frac{\log(s)}{s}}$

Step 2: Calculate $x = x1 * g_1 + m_1$ and $y = y1 * g_1 + m_2$

Step 3: If $x \geq 0$ and $y \geq 0$, then calculate $z = 2x + 3y$ and $w = -x + y$. And if x or y is negative, increment negProb by 1

Step 4: If $z \leq 12$ and $w \leq 1$, then increment region Count by 1

(2) calculate regionProb = regionCount/N.

(3) Calculate 95% confidence interval for the objective mean by $m_1 + 2\ m_2$+/- 1.96* g_1

(4) Multiply the results from B and C and we obtain the 0.95*objectiveProb % confidence interval for the objective mean is $m_1 + 2\ m_2$ +/- 1.96* g_1 and also ≥ 0.

Case 2. If m_1, m_2, and g_1 are unknown, estimate m_1 and m_2 by the sample means. The rest are similar to those in Case 1.

Example 9. Assume X and Y are iid N(2,4). We run 10 times using Monte Carlo algorithm for Example 1. Each run uses N=1000000, the results for both EQ 3.1 and EQ 3.2 are listed in Table 3.

From Step D the 95 * 0.27 = 26% confidence interval for the optimal value falls in 6+/-3.92 or in the interval (2.09, 9.92). From the last row in Table 3, the average of the actual 10 runs matches the result from Step D of the algorithm above. Figure 1 summarizes the results of Table 3.

Table 3. Monte Carlo algorithm for Example 1

Run	1	2	3	4	5	6	7	8	9	10	Average
Max	7.21	7.1	7.37	7.18	6.91	7.24	7.05	7.2	7.21	7.09	7.16
regionProb	0.27	0.23	0.22	0.28	0.21	0.29	0.27	0.24	0.45	0.22	0.27
negProb	0.29	0.27	0.31	0.27	0.34	0.35	0.28	0.26	0.17	0.36	0.29
Actual confidence	0.26	0.26	0.24	0.22	0.25	0.23	0.28	0.31	0.26	0.25	0.26

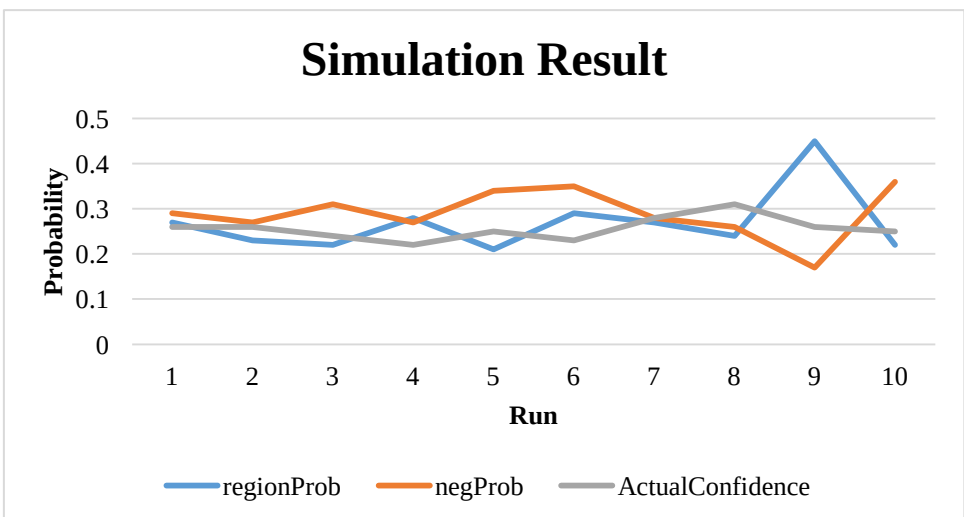

Figure 1. Simulation results for Table 3

6. CONCLUSIONS

The Monte Carlo algorithm proposed in the previous section was based on the assumptions that X and Y are independent and X distributes as $N(m_1 , g_1^2)$ and Y distributes as $N(m_2 , g_1^2)$. However, it can also be used for any discrete distributions with same variances as long as the samples are very large. The fact is based on the Central Limit Theorem[6]. The optimal value 7.40 from the simplex method is very close to the maximum of the objective function value 7.16 from the algorithm. In addition, it falls in the interval (2.09, 9.92) obtained in the last section for Example 1. Although the Monte Carlo algorithm used to solve Example 1 works well, the confidence for creating the interval is extremely low due to the variability of the random variables involved. The differences in the distributions are also intriguing. However, the method can be used in any optimal problem relating to stochastic linear programming.

7. REFERENCES

[1] S. Germano, and D. Yadron, Target faces backlash after 20-day security breach. *Wall Street Journal*, 1(1), p1-2, 2013.

[2] M. Bishop, *Computer security: Art and science*. Boston: Addison-Wesley, 2003.

[3] M. Shing, C. Shing, L. Shing, and L. Shing, Analysis of N category privacy models. *International Journal of Computer Science and Security*, 6(5), p342-358, 2012.

[4] J. Chinneck, *Practical optimization.* Retrieved on September 14, 2014, from http://www.sce.carleton.ca/faculty/chinneck/po.html, 2006.

[5] M. Shing, C. Shing, L. Shing and L. Shing, An optimization approach in information security risk management. *Advances in Management and Applied Economics*, 2(1), p1-12, 2012.

[6] N. Bhat, *Elements of applied stochastic processes*. New Jersey: John Wiley & Sons, 1972.

[7] J. Banks, J. Carson, and B. Nelson, *Discrete event system simulation*. New Jersey: Prentice Hall, 1996.

[8] E. Mendelson, *Introducing game theory and its applications.* New York: Chapman & Hall/CRC Press Co., 2004.

[9] M. Aburdene, *Computer simulation of dynamic systems.* Iowa: Wm. C. Brown Publishing, 1988.