Proc. Estonian Acad. Sci. Phys. Math., 1998, **47**, 1, 19–28
https://doi.org/10.3176/phys.math.1998.1.02

AN APPROXIMATE SOLUTION TO LINEAR AND QUADRATIC PROGRAMMING PROBLEMS BY THE METHOD OF LEAST SQUARES

Evald ÜBI

Tallinn Technical University, Ehitajate tee 5, EE-0026 Tallinn, Estonia; e-mail: evaldy@tv.ttu.ee

Received 6 May 1997, in revised form 28 November 1997

Abstract. The linear programming problem is transformed to the quadratic programming problem. The objective function of the new problem is a sum of squares. To minimize this sum of squares subject to the existing constraints, we use the well-known method of least squares. The quadratic programming problem is solved by the finite orthogonal method, which, compared to the simplex method, uses less high-speed storage, solves ill-conditioned problems more precisely, and is better adjusted for problems with degenerate basis.

Key words: linear and quadratic programming, method of least squares.

1. INTRODUCTION

Suppose a $p \times n$ matrix E, p-vector f, $m \times n$ matrix A, and m-vector b are given. We shall consider the quadratic programming problem of finding a n-vector x^* so that it minimizes the sum of squares

$$z = \| f - Ex \|^2 \to \min$$

subject to

$$Ax = b, \qquad (1)$$
$$x \geq 0,$$

where $\| \cdot \|$ is the Euclidean norm. Assuming $m \leq n$, we examine both cases, $p \leq n$ and $p > n$. Usually, to solve quadratic programming problems, it is recommended to transform the objective function $z = \| f - Ex \|^2$ to the form $z = (c, x) + (x, Dx)$, see [1]. If, however, we square the matrix E, its condition number must also be squared (depends on the definition of the condition number). Our paper recommends the opposite – to transform the function $z = (c, x) + (x, Dx)$ to the sum of squares (see Example 4). The matrix E can be determined by the Cholesky (square root) method and the vector f by solving the triangular system, where rank $D = p$ (p is the number of rows in the matrix E) (see Example 4). The purpose of the paper is to use the advanced least squares techniques in mathematical programming. The most difficult problem arising in this procedure is how to handle the inequalities

$$x \geq 0.$$

We apply the method of least squares to solve the problem (1). To find an approximate solution $x(\epsilon)$ of this problem, we shall present an algorithm VR, using the system

$$\begin{aligned} Ax &= b, \\ \epsilon Ex &= \epsilon f, \\ x &\geq 0, \end{aligned} \tag{2}$$

where ϵ is a small weight, used for the coefficients of the objective function. If $m + p > n$, then the system is overdetermined and $x(\epsilon)$ is its solution in least squares. Let us denote the coefficients of the system (2) by a $m1 \times n$ matrix D and $m1$-vector h, where $m1 = m + p$, $D = (A, \epsilon E)^T$, $h = (b, \epsilon f)^T$, and write the system in the following form

$$\begin{aligned} Dx &= h, \\ x &\geq 0. \end{aligned} \tag{3}$$

We will solve this problem in least squares

$$\Phi\epsilon(x) = \| h - Dx \|^2 \to \min.$$

As $\Phi\epsilon(x) : (\epsilon \times \epsilon) = \| b - Ax \|^2 : (\epsilon \times \epsilon) + \| f - Ex \|^2$, this method is analogous to the penalty function method, see [2]. We will also observe the process of solving the linear programming problem

$$\begin{aligned} z &= (c, x) \to \max, \\ Ax &= b, \\ x &\geq 0, \end{aligned} \tag{4}$$

with the algorithm VR, where c is a n-vector, and $z^* = (c, x^*)$ is the maximum of the objective function. The problem (4) is transformed to the above-mentioned quadratic programming problem

$$\begin{aligned} v &= (z_0 - (c,x))^2 \rightarrow \min, \\ Ax &= b, \\ x &\geq 0, \end{aligned}$$

where z_0 is any number that satisfies the condition $z_0 \geq z^*$. A detailed description of the algorithm VR with examples is given in Sections 2 and 3. Here we shall briefly follow the solving of the problem (3). To find a solution in least squares, we must present the coefficient matrix D as a product of an orthogonal matrix Q and a triangular matrix $R, D = QR$. This QR transformation differs from the usual one in two aspects, see [3]. First, the order of the active variables (i.e., $x_j > 0$) that correspond to the columns of the triangular matrix R, is determined by the third step of the algorithm VR – the variable x_j, for which the column d_j has the minimum angle with the right-hand side h, is activized. Secondly, if in the solution of the system with the triangular matrix R any variable $x_j \leq 0$ (see Example 1, step 4), then the column corresponding to this variable is eliminated from the matrix R and other columns are again transformed to the triangular form by the Givens rotations. After that the order of the matrix R equals to the number of active variables which is one in the first step, two in the second step, etc. In each step their values are determined from the triangular system with the matrix R. In [4], Chapter 22, the problem (1) is examined without the constraints $x \geq 0$, and it is proved that in this case $x(\epsilon) \rightarrow x^*$, if $\epsilon \rightarrow 0$. In the lemma of Section 5 the same is proved if the constraints $x \geq 0$ exist. The algorithm VR is finite, because in each step we find the minimum of the function $\Phi\epsilon(x)$ in a subspace the number of which is finite, see [3].

2. THE ALGORITHM VR OF THE APPROXIMATE SOLUTION TO THE QUADRATIC PROGRAMMING PROBLEMS

Let us describe the algorithm VR for the solution of the problem (1). For this purpose we need the $m1 \times n$ matrix D of the coefficients of the system (3) and the $m1$-vector h, where $m1 = m + p$, n-vectors x, F, G, IJ and $m1$-vector u.

Description of the algorithm VR$(D, h, IJ, x, u, F, G, m1, n, \epsilon, \epsilon 1)$.

1. Initially the number of active (i.e., $x_j > 0$) variables $k = 0$ and $x = 0$.
2. Evaluate the n-vectors F and G with the coordinates

$$F(j) = (d(j), h), \quad G(j) = (d(j), d(j)), \quad j = 1, ..., n.$$

3. Determine the new active variable x_{j0} by computing

$$\max F^2(j)/G(j) = F^2(j_0)/G(j_0) = \text{RE},$$

where the maximum is found for all passive $(x_j = 0)$ variables, for which $F(j) > 0$ and $G(j) > \epsilon 1$.

4. If RE $< \epsilon 1$, then go to step 19.

5. Increase the number of active variables $k = k+1$ and write the index j_0 into the array IJ of active variables.

6. If $k < m1$, apply Householder transformation with a vector $v = d_{j0}$ to the column $d(j)$ and right-hand side h, assuming that these are $m1 - k + 1$ dimensional vectors, see [4], Chapter 10.

7. Compute new $F(j) = F(j) - d(k,j)h(k)$, $G(j) = G(j) - d(k,j)^2$, $j = 1, ..., n$.

8. Solve the upper triangular system of order k to determine the active variables x_j.

9. Initially the number of the controlled variable L equals to $k+1$, $L = k+1$. (During steps 9–12 positivity of active variables is checked).

10. Let $L = L - 1$.

11. If $L = 0$, then go to step 3.

12. If inequality $0 < x_{jL}$ is satisfied, go to step 10 (jL is the index of the active variable).

13. Let $x_{jL} = 0$ and delete the index jL from the massive IJ.

14. From the triangular matrix R we have to eliminate by the Givens rotations the column d_{jL} which corresponds to the active variable x_{jL}. (To annihilate v_2 in a two-dimensional vector $v = (v_1, v_2)^T$, we have to multiply all two-dimensional vectors in these rows, which correspond to v_1 and v_2 by the Givens matrix G, where $G(1,1) = c$, $G(1,2) = s$, $G(2,1) = -s$, $G(2,2) = c$, $c = v_1/\sqrt{(v_1^2 + v_2^2)}$, $s = v_2/\sqrt{(v_1^2 + v_2^2)}$).

15. Displace the indices of active variables, $IJ(i) = IJ(i+1)$ for $i = L, ..., k-1$.

16. Find new $F(j) = F(j) + d(k,j)h(k)$, $G(j) = G(j) + d^2(k,j)$, $j = 1, ..., n$.

17. Reduce the number of active variables, $k = k - 1$.

18. Go to step 8.

19. Check the inequalities $| h(i) | \leq \epsilon 1 + \epsilon(\| b \|^2 + \| f \|^2)$, $i = k+1, ..., m1$. If at least one inequality is not satisfied, the problem has no solution. Stop.

20. Find the minimum sum of squares for the system $Dx = h$, $s^2 = h^2(k+1) + ... + h^2(m1)$.

21. Find the approximate minimum of the objective function $z(\epsilon) = s^2/\epsilon^2$.

22. The problem is solved.

Let us give an example for $\epsilon = 0.3$, $\epsilon 1 = 10^{-25}$.

Example 1.

$$z = (8 - 2x_2)^2 + (5 - x_3)^2 \to \min,$$

$$\begin{aligned} x_1 + 2x_2 + 5x_3 \quad &= 20, \\ 2x_2 - x_3 + x_4 &= 8, \\ x &\geq 0, \end{aligned}$$

for which we have $x^* = (0;\ 3.75;\ 2.5;\ 3)^T$, $z^* = 6.5$.

Iteration		x_1	x_2	x_3	x_4	h
1		1	2	5	0	20
		0	2	–1	1	8
		0	0.6	0	0	2.4
		0	0	0.3	0	1.5
	x	0	0	0	0	
	F	20*	57.4000	92.4500	8	
	G	1*	8.3600	26.0900	1	
2		–1	–2	–5	0	–20
		0	2	–1	1	8
		0	0.6	0	0	2.4
		0	0	0.3	0	1.5
	x	20	0	0	0	
	F	0	17.4400*	–7.5500	8.0000	
	G	0	4.3600*	1.0900	1.0000	
3		–1	–2	–5	0	–20
		0	–2.0881	0.9578	–0.9578	–8.3522
		0	0	0.2873	–0.2873	0
		0	0	0.3	0	1.5
	x	12.0000	4.0000	0	0	
	F	0	0	0.4500*	0.4663	
	G	0	0	0.1726*	0.0826	
4		–1	–2	–5	0	–20.0000
		0	–2.0881	0.9578	–0.9578	–8.3522
		0	0	–0.4154	0.1988	–1.0833
		0	0	0	0.2075	1.0376
	x	–3.4306	5.1961	2.6076	0	
5		0.6917	2.8914	2.7669	0.6917	19.8660
		0.7188	0	4.2935	–0.6787	8.7302
		–0.0699	0	0	–0.1337	0.2397
		0	0	0	0.2075	1.0376
	x	0	4.9250	2.0333	0	
	F	–0.0167	0	0	0.1832*	
	G	0.0049	0	0	0.0609*	
6		0.6917	2.8914	2.7669	0.6917	19.8660
		0.7188	0	4.2935	–0.6787	8.7303
		0.0378	0	0	0.2469	0.7423
		–0.0578	0	0	0	0.7635
	x	0	3.7509	2.5086	3.0069	
	F	–0.0448	0	0	0	
	G	0.0034	0	0	0	

First x_1 is activized ($x_1 = 20$), then x_2 ($x_1 = 12, x_2 = 4$), and x_3. After applying three times Householder transformations, we have $x_1 = -3.4306 < 0$. Then we shall eliminate the first column from the set of active columns. For this end we have to rotate the first and the second row, annihilate the element $d(2,2)$, according to the 14th step of the algorithm VR. After that we will analogously annihilate $d(3,3)$, and then find x_2 and x_3 from the 2×2 system. On the last step x_4 is activized. The approximate solution $x(\epsilon) = (0; 3.7509; 2.5086; 3.0069)$ and $z(\epsilon) = s^2/(\epsilon \times \epsilon) = h_4^2/(\epsilon \times \epsilon) = 0.7635^2/0.3^2 = 6.4776$ that we find will differ only slightly from x^* and z^*.

It is well known that the solution of the problem (3) satisfies the normal equations $D^T D x = D^T h$,

$$\begin{array}{lllll} x_1 & +2x_2 & +5x_3 & & = 20, \\ 2x_1 & +(8+4\epsilon^2)x_2 & +8x_3 & +2x_4 & = 56+16\epsilon^2, \\ 5x_1 & +8x_2 & +(26+\epsilon^2)x_3 & -x_4 & = 92+5\epsilon^2, \\ & 2x_2 & -x_3 & +x_4 & = 8. \end{array}$$

Its non-negative solution in least squares:

$$x(\epsilon) = (0; (390\epsilon^2 + 16\epsilon^4)/t; (260\epsilon^2 + 20\epsilon^4)/t; (312\epsilon^2 + 20\epsilon^4)/t) \to x^*,$$
$$\epsilon \to 0, \quad t = 104\epsilon^2 + 4\epsilon^4. \tag{5}$$

Remark 1. The problem (1) has no solution if $u^* = \min \| b - Ax \|^2 > 0, x \geq 0$. Let k be the number of active variables in the least-squares solution $x(\epsilon)$ of (2). If the right sides $h(k+1), ..., h(m1)$ are close to zero, then the problem (1) is solvable because $h^2(k+1) + ... + h^2(m1) \to u^*$ if $\epsilon \to 0$. The approximate condition of feasibility, given on step 19 of the algorithm, proves the results of calculations. The solvability is determined more surely using the exact algorithm VRA where the function $\phi(x) = \| b - Ax \|^2$ on the set $x \geq 0$ is minimized, see [3].

Remark 2. The criterion for the determination of the activized variable that is used on the third step of the algorithm guarantees maximal linear independence of active columns, because for the dependent columns $F(j) = 0$.

Remark 3. During the actual solution process the equality $x_j > 0$ held almost always.

3. AN APPROXIMATE SOLUTION TO LINEAR PROGRAMMING PROBLEM BY THE ALGORITHM VR

As already mentioned, we can also use the earlier described algorithm VR for solving linear problems. To solve a medium-size linear problem (4), we then have to find a solution to the system

$$\begin{aligned} Ax &= b, \\ \epsilon(c,x) &= \epsilon z_0, \\ x &\geq 0, \end{aligned} \tag{6}$$

in least squares with the aid of the algorithm VR, where z_0 is any number that satisfies the condition $z_0 \geq z^* = (c, x^*)$. An estimate for z_0 is presented in [5]. The less z_0 differs from the maximum z^*, the more precise will be the approximate solution $x(\epsilon)$. Moreover, if $z_0 = z^*$, then $x(\epsilon) = x^*$ for every $\epsilon > 0$. The parameter z_0 must be sufficiently large if we do not know whether the objective function is bounded or not. In the case of a large z_0 we can increase the weight of the constraints by multiplying the equations of the system $Ax = b$ by a sufficiently large number M and this will give us a more exact $x(\epsilon)$. Actually, it gives the same effect as decreasing ϵ, i.e., decreasing the weight of the last equation in systems (2) and (6). We will get the value $z(\epsilon)$ of the objective function, which corresponds to the found approximate solution $x(\epsilon)$,

$$z(\epsilon) = z_0 - \sqrt{(s)}/\epsilon, \tag{7}$$

where s was found on step 20 of the algorithm VR. If the objective function of a linear programming problem is not bounded or if the z_0 chosen is to small (if a feasible solution x_0 exists for which $z_0 = (c, x_0)$), then $z(\epsilon) = 0$.

Example 2.

$$\begin{array}{llll} x_1 & +x_2 & +x_3 & = 3, \\ 2x_1 & & +3x_3 & = 6, \\ x_1 & +3x_2 & +2x_3 & = z \to \max, \\ & & x \geq 0. & \end{array}$$

The solution for the case $\epsilon = 0.01, z_0 = 100$ is given below.

Iteration	x_1	x_2	x_3
1	3.002	0	0
2	2.995	0.034	0
3	–69.751	24.253	48.504
4	0	1.029	1.999

After two Householder and two Givens transformations (see the previous paragraph), the system (6) has the following form

$$\begin{array}{rrrcl} 0.9998x_1 & +1.0000x_2 & +1.0001x_3 & = & 3.0286, \\ 2.0001x_1 & +0.0000x_2 & +3.0000x_3 & = & 5.9969, \\ -0.0133x_1 & +0.0000x_2 & +0.0000x_3 & = & 0.9296. \end{array} \tag{8}$$

We will find the values of active variables x_2 and x_3 from the first two equations, $x(\epsilon) = (0; 1.0290; 1.9994)^T$, $z(\epsilon) = 7.0704$, according to the formula (7). Below the values of $x(\epsilon)$ in the case of different z_0 and ϵ are given.

z_0	ε	x_1	x_2	x_3
10	0.1	0	1.085	1.996
10	0.0001	0	1.000	2.000
100	0.1	0	3.652	1.905
100	0.001	0	1.000	1.999
1000	0.1	0	29.313	0.989
1000	0.001	0	1.003	1.999

The quadratic programming problem (1) observed earlier can be regarded as a multiobjective linear programming problem. The coefficients of the objective function are the rows of the matrix E. The next paragraph examines also the exactness of the solution to linear problems.

4. NUMERICAL EXPERIMENTS

Initially the equations $Ax = b$ must necessarily be present in systems (2) and (3), because otherwise we would have to displace the rows in the systems to quarantee a computing stability, see [4], Chapter 22.

Let us first observe the selection of the weight ϵ. It cannot be too small, because if $\epsilon \to 0$, then the condition number of systems (2) and (6) may converge to infinity, see the formula (5). Moreover, if ϵ is too small, then the system has the same solution as in the case of $\epsilon = 0$ due to the limited accuracy of calculations. This means that we will only get a feasible solution which does not depend on the coefficients of the objective function. As pointed out in [4], Chapter 22, we have to choose ϵ so that beginning from the row $(m+1)$, the absolute values of the nonzero coefficients are substantially smaller than the absolute values of coefficients in the first m rows.

All computations were carried out on an IBM-4381, using FORTRAN codes. For all variables double-precision was used.

Example 3. Let us consider a linear programming problem with the Hilbert matrix, $a(i,j) = 1/(i+j)$, $b(i) = 1/(i+1) + 1/(i+2) + ... + 1/(i+m)$, $c(j) = b(j) + 1/(j+1)$, $i, j = 1, ..., m$. Inequality constraints are transformed to equality constraints with the aid of slack variables. The optimal solution $x_j^* = 1$ was found for $m = 12$, $\epsilon = 0.00001$, $z_0 = 30$ with the accuracy of 0.0001. Well-known programs solve this problem with Hilbert matrix only if m belongs to the interval 4–8.

Example 4.

$$z = -10x_1 + x_1^2 - 2x_1x_2 + 2x_2^2 \to \min,$$

$$\begin{array}{rcl} x_1 + 2x_2 + x_3 & = & 10, \\ x_1 + x_2 \qquad + x_4 & = & 6, \\ x & \geq & 0. \end{array}$$

Transform the problem to the form (1). We shall find the lower triangular matrix R from the equation $R^T R = D$ by using the square root (Cholesky) method. Calculate $r(1,1) = \sqrt{(d(1,1))} = 1$, $r(1,1)r(1,2) = d(1,2) = -1$, $r(1,2) = -1$, $r(2,1) = 0$, $r(2,2)^2 = d(2,2) - r(1,2)^2 = 1$. Determine the quantities f_1 and f_2 from the triangular system $-2f_1 = -10$, $2f_1 - 2f_2 = 0$, the solution of which is $f_1 = f_2 = 5$. The matrix of this system is $-2R^T$. Consequently, the system $Ex = f$ looks as follows:

$$\begin{array}{rcl} x_1 - x_2 & = & 5, \\ x_2 & = & 5, \end{array}$$

where $E = R$. The solution found by the algorithm VR is for $\epsilon = 0.00001$ equal to $(4.600000000330;\ 1.399999999671;\ 2.600000000329;\ 0) = x(\epsilon)$. The Wolfe method uses substantially more high-speed store than algorithm VR, see [1].

Remark 4. The algorithm VR uses less high-speed memory than the quadratic programming algorithms based on the simplex method, see Example 4. The use of orthogonal transformations instead of Gauss's eliminations also enables us to solve the problems more precisely, see Example 3.

5. PROOF OF CONVERGENCE

Lemma. *The solution in least squares of the system* (2) *converges to the solution of the problem* (1), $\lim x(\epsilon) = x^*$, $\epsilon \to 0$.

Proof. Form a system of normal equations. Let $\epsilon \to 0$. In some neighbourhood of the minimum point x^* the active variables are determined from the system of normal equations with the aid of Cramer formulas as a quotient of two polynomials, see the formula (5). These normal equations that correspond to the passive variables $x_j^* = 0$ determine $\Phi\epsilon_j'(x) > 0$ in a certain neighbourhood of the point x^*. There exists $\epsilon_0 > 0$ such that for any positive $\epsilon < \epsilon_0$ the signs of the active variables and the derivative do not change. This means that if having such ϵ, we do not observe the passive variables and normal equations corresponding to them. Then for $\epsilon < \epsilon_0$ all conditions presented in [4], Chapter 22, about the convergence of $x(\epsilon)$ to x^* are satisfied.

The finiteness of the algorithm VR can be concluded from [3].

REFERENCES

1. Künzi, H. und Krelle, W. *Nichtlineare Programmierung*. Springer, Berlin, 1962.
2. Poljak, B. *Vvedenie v optimizatsiyu*. Nauka, Moscow, 1983.
3. Übi, E. Finding non-negative solution of overdetermined or underdetermined system of linear equations by method of least squares. *Trans. Tallinn Technical Univ.*, 1994, **738**, 61–68.
4. Lawson, C. and Hanson, R. *Solving Least Squares Problems*. New-Jersey, 1974.
5. Khachiyan, L. A polynomial algorithm in linear programming. *Doklady AN SSSR*, 1979, **244**, 5, 1093–1096.

LINEAARSE JA RUUTPLANEERIMISE ÜLESANNETE LIGIKAUDNE LAHENDAMINE VÄHIMRUUTUDE MEETODIGA

Evald ÜBI

On esitatud meetod pealkirjas nimetatud ülesannete lahendamiseks, kasutades vähimruutude meetodi tehnikat. Meetodi põhiraskus seisneb muutujate mittenegatiivsuse garanteerimises.