23 LINEAR LEAST SQUARES WITH LINEAR INEQUALITY CONSTRAINTS

Section 1. INTRODUCTION

There are many applications in applied mathematics, physics, statistics, mathematical programming, economics, control theory, social science, and other fields where the usual least squares problem must be reformulated by the introduction of certain inequality constraints. These constraints constitute additional information about a problem.

We shall concern ourselves with linear inequality constraints only. A variety of methods have been presented in the literature. We mention particularly papers that have given serious attention to the numerical stability of their methods: Golub and Saunders (1970), Gill and Murray (1970), and Stoer (1971).

The ability to consider least squares problems with linear inequality constraints allows us, in particular, to have such constraints on the solution as nonnegativity, or that each variable is to have independent upper and lower bounds, or that the sum of all the variables cannot exceed a specified value or that a fitted curve is to be monotone or convex.

Let E be an $m_2 \times n$ matrix, f an m_2-vector, G an $m \times n$ matrix, and h an m-vector. The least squares problem with linear inequality constraints will be stated as follows:

(23.1) PROBLEM LSI

Minimize $\|\mathbf{E}\mathbf{x} - \mathbf{f}\|$ *subject to* $\mathbf{G}\mathbf{x} \geq \mathbf{h}$.

The following important special cases of Problem LSI will also be treated in detail:

(23.2) PROBLEM NNLS (*Nonnegative Least Squares*)

Minimize $||Ex - f||$ *subject to* $x \geq 0$.

(23.3) PROBLEM LDP (*Least Distance Programming*)

Minimize $||x||$ *subject to* $Gx \geq h$.

Conditions characterizing a solution for Problem LSI are the subject of the Kuhn-Tucker theorem. This theorem is stated and discussed in Section 2 of this chapter.

In Section 3 Problem NNLS is treated. A solution algorithm, also called NNLS, is presented. This algorithm is fundamental for the subsequent algorithms to be discussed in this chapter. A Fortran implementation of Algorithm NNLS is given in Appendix C as subroutine **NNLS**.

In Section 4 it is shown that the availability of an algorithm for Problem NNLS makes possible an elegantly simple algorithm for Problem LDP. Besides stating Algorithm LDP in Section 4, a Fortran implementation, subroutine **LDP**, is given in Appendix C.

The problem of determining whether or not a set of linear inequalities $Gx \geq h$ is consistent and if consistent finding some feasible vector arises in various contexts. Algorithm LDP can of course be used for this purpose. The fact that no assumptions need be made regarding the rank of G or the relative row and column dimensions of G may make this method particularly useful for some feasibility problems.

In Section 5 the general problem LSI, having full column rank, is treated by transforming it to Problem LDP. Problem LSI with equality constraints is treated in Section 6.

Finally in Section 7 a numerical example of curve fitting with inequality constraints is presented as an application of these methods for handling constrained least squares problems. A Fortran program, **PROG6**, which carries out this example, is given in Appendix C.

Section 2. CHARACTERIZATION OF A SOLUTION

The following theorem characterizes the solution vector for Problem LSI:

(23.4) THEOREM (*Kuhn-Tucker Conditions for Problem LSI*)

An n-*vector* $\hat{x}$ *is a solution for Problem LSI* (*23.1*) *if and only if there exists an* m-*vector* $\hat{y}$ *and a partitioning of the integers 1 through* m *into subsets* $\mathcal{E}$ *and* $\mathcal{S}$ *such that*

$$G^T\hat{y} = E^T(E\hat{x} - f) \tag{23.5}$$

(23.6) $$\hat{r}_i = 0 \text{ for } i \in \mathcal{E}, \qquad \hat{r}_i > 0 \text{ for } i \in \mathcal{S}$$

(23.7) $$\hat{y}_i \geq 0 \text{ for } i \in \mathcal{E}, \qquad \hat{y}_i = 0 \text{ for } i \in \mathcal{S}$$

where

(23.8) $$\hat{r} = G\hat{x} - h$$

This theorem may be interpreted as follows. Let g_i^T denote the ith row vector of the matrix G. The ith constraint, $g_i^T x \geq h_i$, defines a feasible halfspace, $\{x: g_i^T x \geq h_i\}$. The vector g_i is orthogonal (normal) to the bounding hyperplane of this halfspace and is directed into the feasible halfspace. The point $\hat{x}$ is interior to the halfspaces indexed in $\mathcal{S}$ ($\mathcal{S}$ for *slack*) and on the boundary of the halfspaces indexed in $\mathcal{E}$ ($\mathcal{E}$ for *equality*).

The vector

$$p = E^T(E\hat{x} - f)$$

is the gradient vector of $\varphi(x) = \frac{1}{2}\|Ex - f\|^2$ at $x = \hat{x}$. Since $\hat{y}_i = 0$ for $i \notin \mathcal{E}$ Eq. (23.5) can be written as

(23.9) $$\sum_{i \in \mathcal{E}} \hat{y}_i(-g_i) = -p$$

which states that the negative gradient vector of φ at $\hat{x}$ is expressible as a nonnegative ($\hat{y}_i \geq 0$) linear combination of outward-pointing normals ($-g_i$) to the constraint hyperplanes on which $\hat{x}$ lies ($i \in \mathcal{E}$). Geometrically this means that the negative gradient vector $-p$ lies in the convex cone based at the point $\hat{x}$ and generated by the outward-pointing normals $-g_i$.

Any perturbation u of $\hat{x}$ such that $\hat{x} + u$ remains feasible must satisfy $u^T g_i \geq 0$ for all $i \in \mathcal{E}$. Multiplying both sides of Eq. (23.9) by such a vector u^T and using the fact that the $\hat{y}_i \geq 0$, it follows that u also satisfies $u^T p \geq 0$. From the identity $\varphi(\hat{x} + u) = \varphi(\hat{x}) + u^T p + \|Eu\|^2/2$, it follows that no feasible perturbation of $\hat{x}$ can reduce the value of φ.

The vector $\hat{y}$ (or the negative of this vector) which occurs in the Kuhn-Tucker theorem is sometimes called the *dual* vector for the problem. Further discussion of this theorem, including its proof, will be found in the literature on constrained optimization [see, e.g., Fiacco and McCormick (1968), p. 256].

Section 3. PROBLEM NNLS

Problem NNLS is defined by statement (23.2). We shall state Algorithm NNLS for solving Problem NNLS. The finite convergence of this algorithm will be proved.

We are initially given the $m_2 \times n$ matrix E, the integers m_2 and n, and the m_2-vector f. The n-vectors w and z provide working space. Index sets $\mathcal{P}$

and $\mathcal{Z}$ will be defined and modified in the course of execution of the algorithm. Variables indexed in the set $\mathcal{Z}$ will be held at the value zero. Variables indexed in the set $\mathcal{P}$ will be free to take values different from zero. If such a variable takes a nonpositive value, the algorithm will either move the variable to a positive value or else set the variable to zero and move its index from set $\mathcal{P}$ to set $\mathcal{Z}$.

On termination x will be the solution vector and w will be the dual vector.

(23.10) ALGORITHM NNLS($E, m_2, n, f, x, w, z, \mathcal{P}, \mathcal{Z}$)

Step	*Description*
1	Set $\mathcal{P}$:= NULL, $\mathcal{Z} := \{1, 2, \ldots, n\}$, and $x := 0$.
2	Compute the n-vector $w := E^T(f - Ex)$.
3	If the set $\mathcal{Z}$ is empty or if $w_j \le 0$ for all $j \in \mathcal{Z}$, go to Step 12.
4	Find an index $t \in \mathcal{Z}$ such that $w_t = \max\{w_j : j \in \mathcal{Z}\}$.
5	Move the index t from set $\mathcal{Z}$ to set $\mathcal{P}$.
6	Let $E_{\mathcal{P}}$ denote the $m_2 \times n$ matrix defined by $$\text{Column } j \text{ of } E_{\mathcal{P}} := \begin{cases} \text{column } j \text{ of } E \text{ if } j \in \mathcal{P} \\ 0 \quad \text{if } j \in \mathcal{Z} \end{cases}$$ Compute the n-vector z as a solution of the least squares problem $E_{\mathcal{P}} z \cong f$. Note that only the components $z_j, j \in \mathcal{P}$, are determined by this problem. Define $z_j := 0$ for $j \in \mathcal{Z}$.
7	If $z_j > 0$ for all $j \in \mathcal{P}$, set $x := z$ and go to Step 2.
8	Find an index $q \in \mathcal{P}$ such that $x_q/(x_q - z_q) = \min\{x_j/(x_j - z_j) : z_j \le 0, j \in \mathcal{P}\}$.
9	Set $\alpha := x_q/(x_q - z_q)$.
10	Set $x := x + \alpha(z - x)$.
11	Move from set $\mathcal{P}$ to set $\mathcal{Z}$ all indices $j \in \mathcal{P}$ for which $x_j = 0$. Go to Step 6.
12	*Comment:* The computation is completed.

On termination the solution vector x satisfies

(23.11) $$x_j > 0 \qquad j \in \mathcal{P}$$

(23.12) $$x_j = 0 \qquad j \in \mathcal{Z}$$

and is a solution vector for the least squares problem

(23.13) $$E_{\mathcal{P}} x \cong f$$

The dual vector w satisfies

(23.14) $$w_j = 0 \qquad j \in \mathcal{P}$$

(23.15) $$w_j \leq 0 \qquad j \in \mathcal{Z}$$

and

(23.16) $$w = E^T(f - Ex)$$

Equations (23.11), (23.12), (23.14), (23.15), and (23.16) constitute the Kuhn–Tucker conditions [see Theorem (23.4)] characterizing a solution vector x for Problem NNLS. Equation (23.13) is a consequence of Eq. (23.12), (23.14), and (23.16).

Before discussing the convergence of Algorithm NNLS it will be convenient to establish the following lemma:

(23.17) LEMMA

Let A *be an* m $\times$ n *matrix of rank* n *and let* b *be an* m*-vector satisfying*

(23.18) $$A^Tb = \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ \omega \end{bmatrix} \begin{matrix} \left.\vphantom{\begin{matrix}0\\ \cdot\\ \cdot\\ \cdot\\ 0\end{matrix}}\right\} n-1 \\ \}1 \end{matrix}$$

with

(23.19) $$\omega > 0$$

If $\hat{x}$ *is the least squares solution of* Ax $\cong$ b, *then*

(23.20) $$\hat{x}_n > 0$$

where $\hat{x}_n$ *denotes the* n*th component of* $\hat{x}$.

Proof: Let Q be an $m \times m$ orthogonal matrix that zeros the subdiagonal elements in the first $n - 1$ columns of A, thus

(23.21) $$Q[\underbrace{A}_{n} : \underbrace{b}_{1}] = \begin{bmatrix} \underbrace{\begin{matrix} R \\ 0 \end{matrix}}_{n-1} & \underbrace{\begin{matrix} s \\ t \end{matrix}}_{1} & \underbrace{\begin{matrix} u \\ v \end{matrix}}_{1} \end{bmatrix} \begin{matrix} \}n-1 \\ \}m-n+1 \end{matrix}$$

where R is upper triangular and nonsingular.

Since Q is orthogonal the conditions (23.18) imply

(23.22) $$R^Tu = 0$$

and

$$s^T u + t^T v = \omega > 0 \tag{23.23}$$

Since R is nonsingular, Eq. (23.22) implies that $u = 0$. Thus Eq. (23.23) reduces to

$$t^T v = \omega > 0 \tag{23.24}$$

From Eq. (23.21) it follows that the nth component $\hat{x}_n$ of the solution vector $\hat{x}$ is the least squares solution of the reduced problem

$$t x_n \cong v \tag{23.25}$$

Since the pseudoinverse of the column vector t is $t^T/(t^T t)$, the solution of problem (23.25) can be immediately written as

$$\hat{x}_n = \frac{t^T v}{t^T t} = \frac{\omega}{t^T t} > 0 \tag{23.26}$$

which completes the proof of Lemma (23.17).

Algorithm NNLS may be regarded as consisting of a main loop, Loop A, and an inner loop, Loop B. Loop B consists of Steps 6–11 and has a single entry point at Step 6 and a single exit point at Step 7.

Loop A consists of Steps 2–5 and Loop B. Loop A begins at Step 2 and exits from Step 3.

At Step 2 of Loop A the set $\mathcal{P}$ identifies the components of the current vector x that are positive. The components of x indexed in $\mathcal{Z}$ are zero at this point.

In Loop A the index t selected at Step 4 selects a coefficient not presently in set $\mathcal{P}$ that will be positive [by Lemma (23.17)] if introduced into the solution. This coefficient is brought into the tentative solution vector z at Step 6 in Loop B. If all other components of z indexed in set $\mathcal{P}$ remain positive, then at Step 7 the algorithm sets $x := z$ and returns to the beginning of Loop A. In this process set $\mathcal{P}$ is augmented and set $\mathcal{Z}$ is diminished by the transfer of the index t.

In many examples this sequence of events simply repeats with the addition of one more positive coefficient on each iteration of Loop A until the termination test at Step 3 is eventually satisfied.

However, if some coefficient indexed in set $\mathcal{P}$ becomes zero or negative in the vector z at Step 6, then Step 7 causes the algorithm to remain in Loop B performing a move that replaces x by $x + \alpha(z - x)$, $0 < \alpha \leq 1$, where α is chosen as large as possible subject to keeping the new x nonnegative. Loop B is repeated until it eventually exits successfully at Step 7.

The finiteness of Loop B can be proved by showing that all operations within Loop B are well defined, that at least one more index, the index called q at that point, is removed from set $\mathcal{P}$ each time Step 11 is executed, and that z_t is always positive [Lemma (23.17) applies here]. Thus exit from Loop B at Step 7 must occur after not more than $\pi - 1$ iterations within Loop B, where π denotes the number of indices in set $\mathcal{P}$ when Loop B was entered. In practice Loop B usually exits immediately on reaching Step 7 and does not reach Steps 8—11 at all.

Finiteness of Loop A can be proved by showing that the residual norm function

$$\rho(x) = \|f - Ex\|$$

has a strictly smaller value each time Step 2 is reached and thus that at Step 2 the vector x and its associated set $\mathcal{P} = \{i: x_i > 0\}$ are distinct from all previous instances of x and $\mathcal{P}$ at Step 2. Since $\mathcal{P}$ is a subset of the set $\{1, 2, \ldots, n\}$ and there are only a finite number of such subsets, Loop A must terminate after a finite number of iterations. In a set of small test cases it was observed that Loop A typically required about $\frac{1}{2}n$ iterations.

Updating the QR Decomposition of E

The least squares problem being solved at Step 6 differs from the problem previously solved at Step 6 either due to the addition of one more column of E into the problem at Step 5 or the deletion of one or more columns of E at Step 11. Updating techniques can be used to compute the QR decomposition for the new problem based upon retaining the QR decomposition of the previous problem. Three updating methods are described in Chapter 24. The third of these methods has been used in the Fortran subroutine **NNLS** (Appendix C).

Coping with Finite Precision Arithmetic

When Step 6 is executed immediately after Step 5 the component z_t computed during Step 6 will theoretically be positive. If z_t is not positive, as can happen due to round-off error, the algorithm may attempt to divide by zero at Step 8 or may incorrectly compute $\alpha = 0$ at Step 9.

This can be avoided by testing z_t following Step 6 whenever Step 6 has been entered directly from Step 5. If $z_t \leq 0$ at this point, it can be interpreted to mean that the number w_t computed at Step 2 and tested at Steps 3 and 4 should be taken to be zero rather than positive. Thus one can set $w_t := 0$ and loop back to Step 2. This will result either in termination at Step 3 or the assignment of a new value to t at Step 4.

At Step 11 any x_i whose computed value is negative (which can only be due to round-off error) should be treated as being zero by moving its index from set $\mathcal{P}$ to set $\mathcal{Z}$.

The sign tests on z_i, $i \in \mathcal{P}$, at Steps 7 and 8 do not appear to be critical. The consequences of a possible misclassification here do not seem to be damaging.

A Fortran subroutine **NNLS** implementing Algorithm NNLS and using these ideas for enhancing the numerical reliability appears in Appendix C.

Section 4. PROBLEM LDP

The solution vector for Problem LDP (23.3) can be obtained by an appropriate normalization of the residual vector in a related Problem NNLS (23.2). This method of solving Problem LDP and its verification was brought to the authors' attention by Alan Cline.

We are given the $m \times n$ matrix G, the integers m and n, and the m-vector h. If the inequalities $Gx \geq h$ are compatible, then the algorithm will set the logical variable φ = TRUE and compute the vector $\hat{x}$ of minimal norm satisfying these inequalities. If the inequalities are incompatible, the algorithm will set φ = FALSE and no value will be assigned to $\hat{x}$. Arrays of working space needed by this algorithm are not explicitly indicated in the parameter list.

(23.27) ALGORITHM LDP$(G, m, n, h, \hat{x}, \varphi)$

Step	*Description*
1	Define the $(n + 1) \times m$ matrix E and the $(n + 1)$-vector f as $E := \begin{bmatrix} G^T \\ h^T \end{bmatrix}$ and $f := [\overbrace{0, \ldots, 0}^{n}, 1]^T$. Use Algorithm NNLS to compute an m-vector $\hat{u}$ solving Problem NNLS: Minimize $\lVert Eu - f \rVert$ subject to $u \geq 0$.
2	Compute the $(n + 1)$-vector $r := E\hat{u} - f$.
3	If $\lVert r \rVert = 0$, set φ := FALSE and go to Step 6.
4	Set φ := TRUE.
5	For $j := 1, \ldots, n$, compute $\hat{x}_j := -r_j/r_{n+1}$.
6	The computation is completed.

Proof of Validity of Algorithm LDP

First consider the Problem NNLS, which is solved in Step 1 of Algorithm LDP. The gradient vector for the objective function $\frac{1}{2}\lVert Eu - f \rVert^2$ at the solution point $\hat{u}$ is

$$p = E^T r \tag{23.28}$$

From the Kuhn–Tucker conditions [Theorem (23.4)] for this Problem NNLS there exist disjoint index sets $\mathcal{E}$ and $\mathcal{S}$ such that

(23.29) $$\mathcal{E} \cup \mathcal{S} = \{1, 2, \ldots, m\}$$

(23.30) $$\hat{u}_i = 0 \text{ for } i \in \mathcal{E}, \qquad \hat{u}_i > 0 \text{ for } i \in \mathcal{S}$$

and

(23.31) $$p_i \geq 0 \text{ for } i \in \mathcal{E}, \qquad p_i = 0 \text{ for } i \in \mathcal{S}$$

Using Eq. (23.28) to (23.31) we obtain

(23.32) $$\begin{aligned} \|r\|^2 = r^T r &= r^T[E\hat{u} - f] \\ &= p^T\hat{u} - r_{n+1} = -r_{n+1} \end{aligned}$$

Consider the case in which $\|r\| > 0$ at Step 3. From Eq. (23.32) this implies that $r_{n+1} < 0$, so division by r_{n+1} at Step 5 is valid. Using Eq. (23.31) and (23.32) and the equations of Steps 2 and 5, we establish the feasibility of $\hat{x}$ as follows:

(23.33) $$\begin{aligned} 0 \leq p &= E^T r \\ &= [G : h]\begin{bmatrix} \hat{x} \\ -1 \end{bmatrix}(-r_{n+1}) \\ &= (G\hat{x} - h)\|r\|^2 \end{aligned}$$

Therefore,

(23.34) $$G\hat{x} \geq h$$

From Eq. (23.31) and (23.33) it follows that the rows of the system of inequalities of Eq. (23.34) indexed in set $\mathcal{S}$ are satisfied with equality. The gradient vector for the objective function $\frac{1}{2}\|x\|^2$ of Problem LDP is simply x. The Kuhn–Tucker conditions for $\hat{x}$ to minimize $\frac{1}{2}\|x\|^2$ subject to $Gx \geq h$ require that the gradient vector $\hat{x}$ must be representable as a nonnegative linear combination of the rows of G that are associated with equality conditions in Eq. (23.34), i.e., the rows of G indexed in set $\mathcal{S}$.

From Steps 2 and 5 and Eq. (23.32) we have

$$\begin{aligned} \hat{x} &= \begin{bmatrix} r_1 \\ \cdot \\ \cdot \\ \cdot \\ r_n \end{bmatrix}(-r_{n+1})^{-1} = G^T\hat{u}(-r_{n+1})^{-1} \\ &= G^T\hat{u}\|r\|^{-2} \end{aligned}$$

Noting the sign conditions on $\hat{u}$ given in Eq. (23.30) completes the proof that $\hat{x}$ is a solution of Problem LDP.

It is clearly the unique solution vector since, if $\tilde{x}$ is a different solution vector, then $\|\tilde{x}\| = \|\hat{x}\|$ and the vector $\bar{x} = \frac{1}{2}(\tilde{x} + \hat{x})$ would be a feasible vector having a strictly smaller norm than $\hat{x}$, which contradicts the fact that $\hat{x}$ is a feasible vector of minimum norm.

Now consider the case of $\|r\| = 0$ at Step 3. We must show that the inequalities $Gx \geq h$ are inconsistent. Assume the contrary, i.e., that there exists a vector $\tilde{x}$ satisfying $G\tilde{x} \geq h$. Define

$$q = G\tilde{x} - h = [G : h]\begin{bmatrix} \tilde{x} \\ -1 \end{bmatrix} \geq 0$$

Then

$$\begin{aligned} 0 &= [\tilde{x}^T : -1]r = [\tilde{x}^T : -1]\left\{\begin{bmatrix} G^T \\ h^T \end{bmatrix}\hat{u} - f\right\} \\ &= q^T\hat{u} + 1 \end{aligned}$$

This last expression cannot be zero, however, because $q \geq 0$ and $\hat{u} \geq 0$. From this contradiction we conclude that the condition $\|r\| = 0$ implies the inconsistency of the system $Gx \geq h$. This completes the mathematical verification of Algorithm LDP.

Section 5. CONVERTING PROBLEM LSI TO PROBLEM LDP

Consider Problem LSI (23.1) with the $m_2 \times n$ matrix E being of rank n. In various ways as described in Chapters 2 to 4 one can obtain an orthogonal decomposition of the matrix E:

$$E = Q\begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix}K^T \equiv [\underbrace{Q_1}_{n} : \underbrace{Q_2}_{m_2 - n}]\begin{bmatrix} R \\ 0 \end{bmatrix}K^T \tag{23.35}$$

where Q is $m_2 \times m_2$ orthogonal, K is $n \times n$ orthogonal, and R is $n \times n$ nonsingular. Furthermore, the matrix R can be obtained in triangular or diagonal form.

Introduce the orthogonal change of variables

$$x = Ky \tag{23.36}$$

The objective function to be minimized in Problem LSI can then be written as

(23.37)
$$\varphi(x) = \|f - Ex\|^2 = \left\| \begin{bmatrix} Q_1^T f \\ Q_2^T f \end{bmatrix} - \begin{bmatrix} Ry \\ 0 \end{bmatrix} \right\|^2$$
$$= \|\tilde{f}_1 - Ry\|^2 + \|\tilde{f}_2\|^2$$

where

(23.38)
$$\tilde{f}_i = Q_i^T f \qquad i = 1, 2$$

With a further change of variables,

(23.39)
$$z = Ry - \tilde{f}_1$$

we may write

(23.40)
$$\varphi(x) = \|z\|^2 + \|\tilde{f}_2\|^2$$

The original problem LSI of minimizing $\|f - Ex\|$ subject to $Gx \geq h$ is thus equivalent, except for the additive constant $\|\tilde{f}_2\|^2$ in the objective function, to the following Problem LDP:

(23.41)
$$\text{Minimize } \|z\|$$
$$\text{subject to } GKR^{-1}z \geq h - GKR^{-1}\tilde{f}_1$$

If a vector $\hat{z}$ is computed as a solution of this Problem LDP, then a solution vector $\hat{x}$ for the original Problem LSI can be computed from Eq. (23.39) and (23.36). The squared residual vector norm for the original problem can be computed from Eq. (23.40).

Section 6. PROBLEM LSI WITH EQUALITY CONSTRAINTS

Consider Problem LSI (23.1) with the addition of a system of equality constraints, say $C_{m_1 \times n}x = d$, with Rank$(C) = m_1 < n$ and Rank $([C^T : E^T]) = n$. The equality constraint equations can be eliminated initially with a corresponding reduction in the number of independent variables. Either the method of Chapter 20 or that of Chapter 21 is suitable for this purpose.

Using the method of Chapter 20, introduce the orthogonal change of variables

(23.42)
$$x = K \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \begin{matrix} \}m_1 \\ \}n - m_1 \end{matrix}$$

where K triangularizes C from the right:

(23.43)
$$\begin{bmatrix} C \\ E \\ G \end{bmatrix} K = \begin{bmatrix} \tilde{C}_1 & 0 \\ \tilde{E}_1 & \tilde{E}_2 \\ \tilde{G}_1 & \tilde{G}_2 \end{bmatrix}$$
$$\quad \underbrace{}_{m_1} \quad \underbrace{}_{n-m_1}$$

Then $\hat{y}_1$ is determined as the solution of the lower triangular system $\tilde{C}_1 y_1 = d$, and $\hat{y}_2$ is the solution of the following Problem LSI:

(23.44)
$$\text{Minimize } \|\tilde{E}_2 y_2 - (f - \tilde{E}_1\hat{y}_1)\|$$
$$\text{subject to } \tilde{G}_2 y_2 \geq h - \tilde{G}_1 y_1$$

After solving problem (23.44) for $\hat{y}_2$ the solution $\hat{x}$ can be computed using Eq. (23.42).

If the method of Chapter 21 is used, one would compute $Q_1, \tilde{C}_1, \tilde{C}_2, \tilde{d}, \tilde{E}_1, \tilde{E}_2$, and $\tilde{f}$ using Eq. (21.11) to (21.14) and additionally solve for the matrix $\tilde{G}_1$ in the upper triangular system

(23.45)
$$\tilde{G}_1\tilde{C}_1 = G_1$$

Then $\hat{x}_2$ is the solution of the following Problem LSI:

(23.46)
$$\text{Minimize } \|\tilde{E}_2 x_2 - \tilde{f}\|$$
$$\text{subject to } (G_2 - \tilde{G}_1\tilde{C}_2)x_2 \geq h - \tilde{G}_1\tilde{d}$$

and $\hat{x}_1$ would be computed by solving the upper triangular system

(23.47)
$$\tilde{C}_1 x_1 = \tilde{d} - \tilde{C}_2\hat{x}_2$$

Section 7. AN EXAMPLE OF CONSTRAINED CURVE FITTING

As an example illustrating a number of the techniques that have been described in this chapter we consider a problem of fitting a straight line to a set of data points where the line must satisfy certain constraints. **PROG6**, a Fortran main program that performs the computation for this example, is given in Appendix C.

Let the data be given as follows:

t	w
0.25	0.5
0.50	0.6
0.50	0.7
0.80	1.2

We wish to find a line of the form

$$f(t) = x_1 t + x_2 \tag{23.48}$$

which fits these data in a least squares sense subject to the constraints

$$f'(t) \geq 0 \tag{23.49}$$

$$f(0) \geq 0 \tag{23.50}$$

$$f(1) \leq 1 \tag{23.51}$$

This problem can be written as Problem LSI:

$$\begin{aligned} &\text{Minimize } \|Ex - f\| \\ &\text{subject to } Gx \geq h \end{aligned} \tag{23.52}$$

where

$$E = \begin{bmatrix} 0.25 & 1 \\ 0.50 & 1 \\ 0.50 & 1 \\ 0.80 & 1 \end{bmatrix}, \qquad f = \begin{bmatrix} 0.5 \\ 0.6 \\ 0.7 \\ 1.2 \end{bmatrix}$$

$$G = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & -1 \end{bmatrix}, \qquad h = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}$$

We compute an orthogonal decomposition of the matrix E in order to convert the Problem LSI to a Problem LDP as described in Section 5 of this chapter. Either a QR or a singular value decomposition of E could be used. We shall illustrate the use of a singular value decomposition.

$$E = U_{4\times 4}\begin{bmatrix} S_{2\times 2} \\ 0_{2\times 2} \end{bmatrix} V^T_{2\times 2}$$

$$S = \begin{bmatrix} 2.255 & 0.0 \\ 0.0 & 0.346 \end{bmatrix}$$

$$V = \begin{bmatrix} -0.467 & 0.884 \\ -0.884 & -0.467 \end{bmatrix}$$

$$\begin{matrix} 2\{ \\ 2\{ \end{matrix}\begin{bmatrix} \tilde{f}_1 \\ \tilde{f}_2 \end{bmatrix} = U^T f = \begin{bmatrix} -1.536 \\ 0.384 \\ -0.054 \\ 0.174 \end{bmatrix}$$

Introduce the change of variables

$$z = SV^Tx - \tilde{f}_1 \tag{23.53}$$

We then wish to solve the following Problem LDP:

$$\begin{aligned} &\text{Minimize } \|z\| \\ &\text{subject to } \tilde{G}z \geq \tilde{h} \end{aligned} \tag{23.54}$$

where

$$\tilde{G} = GVS^{-1} = \begin{bmatrix} -0.207 & 2.558 \\ -0.392 & -1.351 \\ 0.599 & -1.206 \end{bmatrix}$$

and

$$\tilde{h} = h - \tilde{G}\tilde{f}_1 = \begin{bmatrix} -1.300 \\ -0.084 \\ 0.384 \end{bmatrix}$$

A graphical interpretation of this Problem LDP is given in Fig. 23.1. Each row of the augmented matrix $[\tilde{G} : \tilde{h}]$ defines one boundary line of the feasible region. The solution point $\hat{z}$ is the point of minimum euclidean norm

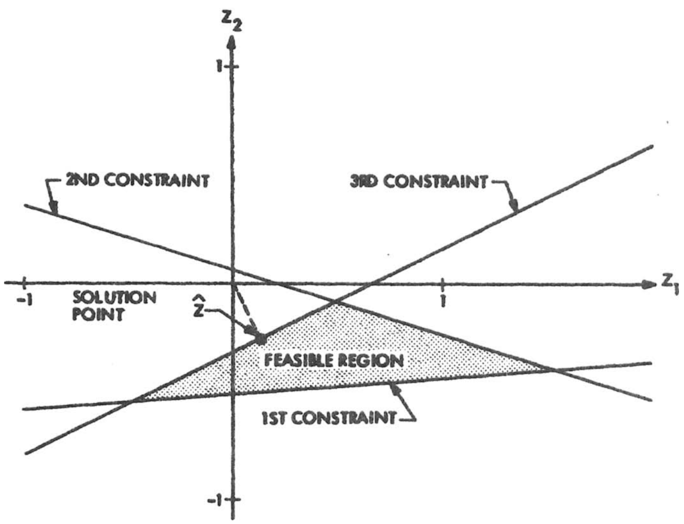

Fig. 23.1 Graphical interpretation of the sample Problem LDP (23.54).

within the feasible region. This point, as computed by subroutine LDP, is

$$\hat{z} = \begin{bmatrix} 0.127 \\ -0.255 \end{bmatrix}$$

Then using Eq. (23.53) we finally compute

$$\hat{x} = VS^{-1}(\hat{z} + \tilde{f}_1) = \begin{bmatrix} 0.621 \\ 0.379 \end{bmatrix}$$

The residual vector for the solution vector $\hat{x}$ is

$$\hat{r} = f - E\hat{x} = \begin{bmatrix} -0.034 \\ -0.089 \\ 0.011 \\ 0.324 \end{bmatrix}$$

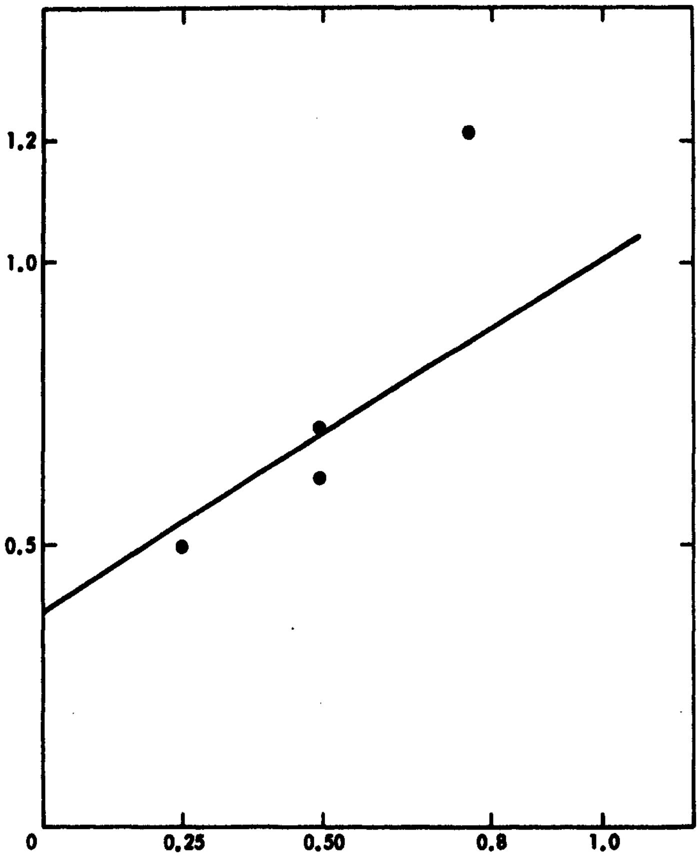

Fig. 23.2 Graph of solution line for the sample problem (23.48)–(23.51).

and the residual norm is

$$\|\hat{r}\| = 0.338$$

The given data points (t_i, w_i), $i = 1, \ldots, 4$, and the fitted line, $f(t) = 0.621t + 0.379$, are shown in Fig. 23.2. Note that the third constraint, $f(1) \leq 1$, is active in limiting how well the fitted line approximates the data points.

The numerical values shown in describing this example were computed using a UNIVAC 1108 computer. Executing the same Fortran code on an IBM 360/67 resulted in opposite signs in the intermediate quantities V, $\tilde{f}_1$, $\tilde{G}$, and $\hat{z}$. This is a consequence of the fact that the signs of columns of the matrix V in a singular value decomposition are not uniquely determined. The difference in the number of iterations required to compute the singular value decomposition of the matrix E on the two computers having different word lengths resulted in a different assignment of signs in the matrices U and V.

EXERCISES

(23.55) Prove that if a Problem LSE has a unique solution *without* inequality constraints, then it has a unique solution *with* inequality constraints.

(23.56) Show that the problem of minimizing a quadratic function $f(x) = \frac{1}{2}x^TBx + a^Tx$, for positive definite B can be transformed to the problem of minimizing $\frac{1}{2}\|w\|^2$ by letting $w = Fx - g$ for an appropriate choice of the nonsingular matrix F and vector g.

(23.57) If the function f of Ex. (23.56) is to be minimized subject to the constraints $Cx = d$ and $Gx \geq h$, what are the corresponding constraints for the problem of minimizing $\frac{1}{2}\|w\|^2$?