Mathematical programming in statistical estimation theory

Ronald Raymond Hocking

Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/rtd

Part of the Mathematics Commons

Recommended Citation


https://lib.dr.iastate.edu/rtd/2297

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
This dissertation has been microfilmed exactly as received

HOCKING, Ronald Raymond, 1932–
MATHEMATICAL PROGRAMMING IN STATISTICAL ESTIMATION THEORY.

Iowa State University of Science and Technology
Ph.D., 1962
Mathematics

University Microfilms, Inc., Ann Arbor, Michigan
MATHEMATICAL PROGRAMMING IN STATISTICAL ESTIMATION THEORY

by

Ronald Raymond Hocking

A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of
The Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Major Subjects: Statistics
Mathematics

Approved:

Signature was redacted for privacy.

In Charge of Major Work

Signature was redacted for privacy.

Heads of Major Departments

Signature was redacted for privacy.

Dean of Graduate College

Iowa State University
Of Science and Technology
Ames, Iowa

1962
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>INTRODUCTION</strong></td>
<td>1</td>
</tr>
<tr>
<td><strong>PART I. MATHEMATICAL PROGRAMMING</strong></td>
<td>4</td>
</tr>
<tr>
<td>REVIEW OF LITERATURE</td>
<td>5</td>
</tr>
<tr>
<td>A NEW METHOD OF CONVEX PROGRAMMING</td>
<td>12</td>
</tr>
<tr>
<td><strong>PART II. THE APPLICATION OF MATHEMATICAL PROGRAMMING TO STATISTICAL</strong></td>
<td>37</td>
</tr>
<tr>
<td>ESTIMATION THEORY</td>
<td></td>
</tr>
<tr>
<td>REVIEW OF LITERATURE</td>
<td>38</td>
</tr>
<tr>
<td>THE THEORY OF LEAST SQUARES ESTIMATION WITH</td>
<td>41</td>
</tr>
<tr>
<td>CONSTRAINED PARAMETERS</td>
<td></td>
</tr>
<tr>
<td>THE DISTRIBUTION OF THE PROJECTED LEAST SQUARES ESTIMATOR</td>
<td>58</td>
</tr>
<tr>
<td>A SPECIAL QUADRATIC PROGRAMMING PROCEDURE</td>
<td>77</td>
</tr>
<tr>
<td><strong>LITERATURE CITED</strong></td>
<td>100</td>
</tr>
<tr>
<td><strong>ACKNOWLEDGMENTS</strong></td>
<td>103</td>
</tr>
<tr>
<td><strong>APPENDIX A. THE MODIFIED SIMPLEX METHOD AND THE</strong></td>
<td>104</td>
</tr>
<tr>
<td>PROJECT FORM OF THE INVERSE</td>
<td></td>
</tr>
<tr>
<td><strong>APPENDIX B. SUMMARY OF CALCULATIONS FOR</strong></td>
<td>108</td>
</tr>
<tr>
<td>EXAMPLE 1, PART I</td>
<td></td>
</tr>
<tr>
<td><strong>APPENDIX C. SOME GEOMETRY OF SURFACES IN</strong></td>
<td>117</td>
</tr>
<tr>
<td>p-DIMENSIONS</td>
<td></td>
</tr>
</tbody>
</table>
INTRODUCTION

The concept of mathematical programming covers a wide range of techniques used in various fields for determining the maximum or minimum of a mathematical function of many variables when, due to limits on the resources, the solution is constrained to lie in a subspace of the variable space. Assuming that a mathematical model has been designed to fit a particular situation, there still remains the problem of finding a method of analysis, usually taking advantage of special characteristics of the problem under consideration. Many problems in economics as well as in other fields fall into the special category known as linear programming, and as a result the literature in this area is quite extensive. Mathematically, the linear programming problem can be stated as that of finding the optimum value (maximum or minimum) of a linear function of several variables, referred to as the objective function, subject to the condition that certain linear inequalities involving these variables (also called activities) must be satisfied. One of the main reasons for the extensive use of the linear programming concept has been the development of the simplex method. During the past decade, this method has been extended and modified to the point where it is now a very efficient tool for solving a wide variety of linear problems.
Of course, many problems in determining an optimum subject to certain constraints cannot be stated in terms of linear functions, thus giving rise to the concept of nonlinear programming. Thus, we may generalize the above linear problem to the case where the objective function and the restraint inequalities may be nonlinear functions. Attempts to solve the nonlinear programming problem with complete generality have not met with much success and so attention had been focused on special cases.

Many nonlinear problems lead to an objective function which is a quadratic or can be approximated by a quadratic. Solutions to such problems, at least for the case of linear restrictions, are numerous and computationally all of them are quite simple.

A much more general problem is that known as convex programming. In this situation one is concerned with finding, say, the minimum value of a convex function subject to the condition that the solution is constrained to lie in a convex region. An important feature of this problem which makes it amenable to mathematical and numerical treatment is the fact that local minima are in fact global minima.

In this dissertation we are interested in two main problems. The first of these is to find a computationally simple solution to the convex programming problem. The algorithm proposed here has several advantages over existing
methods, the most important being that, with a few small modifications, the existing computer codes for the modified simplex method can be used.

The second problem with which we are concerned is that of estimating the parameters in a linear regression model when these parameters are constrained to lie in a convex region of the parameter space. Computationally, the problem can be formulated as a problem in convex programming, in fact, the constrained minimization of a quadratic function.

Certain properties of this quadratic programming problem suggest an alternative algorithm. This algorithm, although computationally more difficult than the general convex programming algorithm, has certain desirable features and in addition it can be applied to a slightly larger class of restraining functions.

Some of the properties of the estimators are investigated with particular attention given to asymptotic properties. For certain special cases, the exact probability distribution of the estimators is derived.
PART I. MATHEMATICAL PROGRAMMING
Since we shall be using many of the properties of the linear programming problem, we begin with a formal statement of this problem and a review of some of the relevant theory. For a more detailed discussion we refer to Charnes and Cooper [7]*. In the discussions that follow we find it convenient to use matrix notation except where there is a possibility of confusion. We adopt the convention that all vectors are column vectors and use a prine (') to denote the corresponding row vector. With this notation, we state the linear programming problem as that of finding the vector $x$ which will maximize the linear objective function

$$c'x$$

subject to the conditions

$$Ax \leq b$$

$$x \geq 0$$

Here $b$ and $c$ are $n \times 1$ vectors of known constants, $x$ is an $n \times 1$ vector whose values are to be determined, and $A$ is an $m \times n$ matrix of known constants.

*Hereafter brackets [ ] will be used to refer to literature cited and parentheses ( ) will be used to denote equations, inequalities, etc.
The same data used to express the maximization problem given by (1) and (2) may be used to formulate another problem called the dual linear programming problem. This problem is stated as the minimization of the linear function

\[ b'w \]  \hspace{1cm} (3) \]

subject to the conditions

\[ A'w \geq c \]
\[ w \geq 0 \]  \hspace{1cm} (4)

Here A, b, and c are as defined above and w is an m x 1 vector to be determined. It is customary to refer to the maximization problem as the direct or primal problem and to the minimization problem as the dual problem. Which problem is called the direct and which the dual is of no consequence since, for example, the minimization of b'w is equivalent to the maximization of (-b)'w and the condition A'w \geq c is equivalent to (-A)'w \leq -c. In fact, in a later section we shall express both problems in terms of a maximization.

The fundamental relations between the linear programming problem and its dual are given by the Dual Theorem of Linear Programming [see, e.g., Charnes and Cooper, 7, p. 182]. This theorem states that the direct linear programming problem has a finite solution if and only if its dual has a finite solution, and that the optimum values are equal. When either of
the problems has no solution then the other has no solution or an infinite solution. Also, if one problem has an infinite solution, then the other has no solution.

The methods of linear programming typically solve both the dual and the direct problem simultaneously. Thus, for example, in the simplex method we have available after the final iteration of the direct problem, the solution to both the direct and the dual problems with no extra work. We shall see later, however, that although the two problems are equivalent, solving the dual rather than the direct problem is the key to the convex programming algorithm.

We shall not discuss the simplex method here but merely recall that the method is based on the investigation of extreme points of the region defined by (2), called the feasible region. This approach is equivalent to investigating sets of vectors selected from the columns of the matrix A (in the direct problem) and the m unit (or slack) vectors which form a basis for the m-dimensional Euclidean space. The simplex calculations proceed from one basis to another in such a way that the value of the functional $c'x$, sometimes referred to as the "direct profit", is monotonically nondecreasing. In Appendix A we discuss briefly the modified simplex algorithm which we shall use in our convex programming procedure.

The literature on nonlinear programming is quite extensive and so we shall confine ourselves here to reviewing
only a selection of some of the papers which are most relevant to the present method. There are, of course, numerous papers devoted to particular problems which we shall not discuss.

One of the basic theoretical papers on nonlinear programming was presented by Kuhn and Tucker [23]. Their paper extends the notion of equivalence between linear programming and saddle value problems [see Goldman and Tucker, 14] to the nonlinear programming problem. In order to discuss the important Equivalence Theorem of Kuhn and Tucker we give a brief statement of the convex programming problem. We are interested in finding an m-dimensional vector \( x^0 \) which maximizes the concave function \( g(x) \) subject to the restrictions \( \sum_{i=1}^{m} f_i(x) \leq 0 \), \( i = 1 \ldots m \), and \( x_j \geq 0 \), \( j = 1 \ldots n \), where the \( f_i(x) \) are convex functions. These restrictions define a convex region hereafter referred to as the feasible region. If we now consider the Lagrange function

\[
L(x, w) = g(x) + \sum_{i=1}^{m} w_i f_i(x)
\]

then the Equivalence Theorem says that, under certain regularity conditions, \( x^0 \) is a solution to the maximization problem if and only if \( x^0 \) and some vector \( w^0 \) are a saddle-point of the function \( L(x, w) \). That is to say, for all \( x \geq 0 \), \( w \geq 0 \) we have
We may also write this in the following manner:

\[
L(x^0, w^0) = \min_{w \geq 0} \max_{x \geq 0} L(x, w) = \max_{x \geq 0} \min_{w \geq 0} L(x, w)
\]  

Based on this theorem, Arrow et al. [4] have developed a gradient method for determining the saddle-point of a function and hence solving the convex programming problem. This method consists of solving a system of differential equations at each stage of an iterative process. The solutions of these differential equations are shown to converge to the saddle-point in a large number of differentiably small steps.

Numerous other algorithms have been based on steepest ascent methods. Dennis [12] and Zoutendijk [33] have independently devised a method of "feasible directions". This is an iterative method which at each stage determines (a) a direction in which the objective function increases and the trial solution remains feasible, and (b) the length of the step to be taken in that direction. The determination of the "best" direction requires the solution of a quadratic programming problem and the length of the step is determined by solving the one-dimensional maximization problem in the given direction with the stipulation that the trial solution remains
Closely related to the above method is the "gradient projection" method of Rosen [28]. In this method one proceeds in the direction of the gradient of the objective function when possible, otherwise, the gradient is projected onto the supporting hyperplanes of the convex region at the current trial point and a step is taken in the direction of the projected gradient. The new trial solution is then not feasible and a procedure is given to correct back to the feasible region in such a way that a monotone sequence of functional values is obtained.

It should be emphasized that the necessity of maintaining feasibility at each stage of the iteration causes a considerable amount of additional computation in both of the above methods. In direct contrast to these methods is the "cutting-plane" method of Kelley [21]. This method, which was developed independently by Cheney and Goldstein [11], approaches the solution from outside the feasible space and thus avoids cumbersome procedures for keeping the trial point feasible. On the other hand it requires the solution of an infinite sequence of linear programs with some uncertainty as to how one keeps the number of restrictions from becoming very large.

Wegner [31] has devised a nonlinear extension of the simplex method which introduces coefficients analogous to the
elements of the matrix $A$ in the linear problem discussed earlier. The amount of computation is increased considerably by the necessity of solving sets of nonlinear simultaneous equations at each iteration.

The solution presented here for the convex programming problem is most closely related in its geometrical interpretation to the "cutting-plane" method. However, it arises from a completely different principle, namely as the dual to the linear programming problem for which the tangential planes to the original feasible region form the boundaries. Consequently there results a much simplified algorithm closely related to the modified simplex method employing the product form of the inverse. Moreover, the present method only requires the solution of one linear programming problem in which a set of $n + 1$ column vectors of $n + 1$ elements form the basis matrix. This procedure is like that presented by Hartley [16] for the special case of "separable" functions in that it is based on solving the dual problem by a modified simplex algorithm.
A NEW METHOD OF CONVEX PROGRAMMING

Formulation of the problem

We consider the following statement of the convex programming problem:

Maximize $g(x)$

subject to the restrictions

$$\sum_{i=1}^{m} f(x) \leq \sum_{i=1}^{m} b_i \quad i = 1 \ldots m \quad (9)$$

where $x$ is an $n \times 1$ vector, $f(x), f(x), \ldots, f(x)$ and $-g(x)$ are differentiable, convex functions defined for all $x$, and $b, \ldots, b$ are known constants. We assume that the region defined by (9) is bounded. The restriction of convexity of the $f(x)$ and $-g(x)$ implies that for any two vectors $x^1$ and $x^2$, and for any $0 \leq \gamma \leq 1$ that

$$f[\gamma x^1 + (1 - \gamma)x^2] \leq \gamma f(x^1) + (1 - \gamma)f(x^2) \quad (10)$$

The problem defined by (8) and (9) is unchanged if we introduce the functional as an additional restriction, i.e.
\[ x_{n+1} = g(x) \quad (11) \]

and maximize the coordinate \( x_{n+1} \). The equality restriction given by (11) is equivalent to the two inequality restrictions

\[
g(x) - x_{n+1} \leq 0 \quad (12)
\]

\[
-g(x) + x_{n+1} \leq 0 \quad (13)
\]

Now it is easily seen that the problem of maximizing \( x_{n+1} \) subject to the restrictions (9), (12) and (13) is equivalent to maximizing \( x_{n+1} \) subject to (9) and (13). The additional restriction \( x_{n+1} \geq 0 \) can be introduced since maximizing \( g(x) \) plus a constant does not change the original problem. Note also that the function \( x_{n+1} - g(x) \) is convex.

We now replace each of the convex restrictions by a set of linear restrictions. We do this by imposing a grid on the \( n \)-dimensional Euclidean space and constructing the tangent planes to the surface \( z = f(x) \) at each grid point. Thus we replace the single nonlinear restriction \( f(x) \leq b \) by the set of linear inequalities

\[
i^f(x^*) + \sum_{j=1}^{n} i^f_j (x_j - x_j^*) \leq b \quad (14)
\]

where \( i^f_j = \frac{\delta i^f}{\delta x_j} \) evaluated at the grid point \( x^* \). Because of
the convexity of \( f(x) \) the polyhedral space defined by (14) contains the convex space \( f(x) \leq b \) entirely. As the grid width of the \( x^* \) tends to zero the intersection of the polyhedral spaces (14) for \( i = 1, 2, \ldots, m \) tends to that described by (9). Similarly, the restriction \(-g(x) + x_{n+1} \leq 0\) is replaced by the set of linear inequalities:

\[
-g(x^*) - \sum_{j=1}^{n} g^*_j (x_j - x^*_j) + x_{n+1} \leq 0.
\]  

(15)

For convenience we summarize (14) and (15) in matrix notation as follows:

\[
iAx \leq i^c \quad i = 1, 2, \ldots, m
\]  

(16)

with \( i^c = b - if(x^*) + \sum_{j=1}^{n} if^*_j x^*_j \)

\[
m+1Ax + x_{n+1} e \leq m+1^c
\]  

(17)

with \( m+1^c = g(x^*) - \sum_{j=1}^{n} g^*_j x^*_j \)

Here the matrices \( iA \) have as elements the partial derivatives of \( f \) evaluated at the grid points while the matrix \( m+1A \) comprises the partial derivative of \(-g(x)\) at the same grid points and \( e \) is a column vector of ones.

Since the region defined by (9) is bounded we are able
to postulate a large cube defined by $|x_j| < D$ completely containing this region. Tangent planes to any of the $f(x)$ and $-g(x)$ will only be introduced for such grid points $x^*$ with $|x_j^*| < D$ and hence (16) and (17) represent a finite, but large number (say $N$) of linear inequalities. To these inequalities we adjoin the $2n$ additional restrictions $x_j \leq D$ and $-x_j \leq D$.

The maximization of $x_{n+1}$ subject to these $N + 2n$ linear restrictions and the additional restrictions $x_j \geq 0$ $j = 1 \ldots n+1$ is thus a finite linear problem and all of the existing theory applies. For a fine grid, this finite problem is an approximation to the original problem defined by (8) and (9) and it is this finite problem which we propose to solve. The task of solving this problem for small grid size is clearly unmanageable by the usual methods but we shall develop an algorithm to overcome this difficulty.

**The dual tableau**

By application of the dual theorem of linear programming we see that if the linear problem as defined above has a solution, then this solution can be obtained by solving the dual problem. In the above notation, this problem can be stated as follows: (For simplicity we temporarily ignore the restrictions $x_j \leq D$, $-x_j \leq D$, $j = 1, 2, \ldots, n$.)

Maximize $-w' c$  \hspace{1cm} (18)
subject to the linear restrictions which we express in terms of a partitioned matrix as:

\[
\begin{bmatrix}
-1A' & \cdots & -mA' & -(m+1)A' \\
- & - & - & - \\
0 & \cdots & 0 & -e'
\end{bmatrix}
\begin{align*}
w & \leq -d \quad (19) \\
w & \geq 0.
\end{align*}
\]

Here the column vector \( c \) is the vector of right-hand sides of (16) and (17). That is, it may be written as a partitioned vector, \( c' = (\gamma c', \beta c', \cdots, m+1c') \). The \( iA' \) are the transposes of the matrices occurring in (16) and (17) and \( d \) is an \( n+1 \times 1 \) column vector with a one in the \( (n+1) \)st position and zeros elsewhere.

The dual tableau is displayed in Tableau 1. It is constructed as follows:

(a) The first \( n \) rows of the coefficient matrix in (19) are placed in the body of the tableau as lines \( j = 1 \) to \( n \). For each of these a column slack vector \( S_j \) is introduced.

(b) The \( (n+1) \)st row of the matrix in (19) is multiplied by \(-1\) and entered into the tableau as line \( n+1 \). A column excess vector \( E \) and an artificial vector \( A \) are added.

(c) The vector \( d \) is placed in the column labeled \( \text{P} \).
Tableau 1. The dual tableau

<table>
<thead>
<tr>
<th>j</th>
<th>P_0</th>
<th>(i^{th}) set of restrictions</th>
<th>(m+1)^{st} set</th>
<th>E</th>
<th>A</th>
<th>S_1 ... S_n</th>
<th>S_0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>(i^c)</td>
<td>(m+1^c)</td>
<td>0</td>
<td>M</td>
<td>0 ... 0</td>
<td>1</td>
</tr>
<tr>
<td>l</td>
<td>0</td>
<td>(-\frac{\partial f}{\partial x_1})</td>
<td>(\frac{\partial g}{\partial x_1})</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>n</td>
<td>0</td>
<td>(-\frac{\partial f}{\partial x_n})</td>
<td>(\frac{\partial g}{\partial x_n})</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>n+1</td>
<td>1</td>
<td>0 ... 0</td>
<td>1 ... 1</td>
<td>-1</td>
<td>1</td>
<td>0 ... 0</td>
<td>0</td>
</tr>
</tbody>
</table>
(d) The objective function which consists of the vector $c$ and a number $M$ in column $A$ is placed in line $j = 0$. The slack vector $S_0$ is added for the objective function.

(e) The effect of the restrictions $x_j \leq D$ and $-x_j \leq D$, $j = 1 \ldots n$ in the direct problem is to introduce $2n$ additional columns in the dual tableau which are identical to $S_1 \ldots S_n$ or $-S_1 \ldots -S_n$ with the exception that they have the number $D$ in the row $j = 0$. These vectors are omitted from Tableau 1 for simplicity, and in practice they are only conceptually in the tableau since in general it will not be necessary to specify the value for $D$. (The role of these columns will be made clear when we discuss the algorithm.)

The algorithm

The modified simplex method using the product form of the inverse [see Orchard-Hays, 25] is now applied to Tableau 1. The important feature of the procedure which follows is the fact that it is not necessary to explicitly form all of the columns of the tableau.

If a vector $(x_1 \ldots x_n)$ is known for which $\frac{a}{a_j} x_j \leq 0$ $j = 1 \ldots n$, then this vector with the slack vectors $S_1$, $S_2 \ldots S_m$ can be used to give a basic feasible solution to start the problem, and the artificial vector $A$ is not necessary. The "pricing vector", $(1, p_1 \ldots p_{n+1})$ is computed and the "pricing operation" [Orchard-Hays, 25] is performed as
described below. If such a vector is not available, the artificial vector, A, must be retained to give an initial basic feasible solution. The determination of the value for M must be made from a prior knowledge of the problem. This number must be so large that it is inconceivable that the vector A remain in the basis at the termination of the problem, or in terms of the direct problem, the value for M is such that the maximum value of $x_{n+1}$ subject to the restrictions in the direct problem could not possibly exceed M.

In either case, if we consider the pricing vector applied to the columns of the tableau at any stage, we have as the net price for any column (other than the slack and artificial vectors):

$$i^c* - \sum_{j=1}^{n} p_j i^j f^*$$

$$i = 1 \ldots m \quad (20)$$

or

$$m+1^c* + \sum_{j=1}^{n} p_j g^* + p_{n+1}$$

$$\quad (21)$$

where by definition,

$$i^c* = i^b - i f(x^*) + \sum_{j=1}^{n} i^j f^* x^*$$

$$i = 1 \ldots m \quad (22)$$

and
\[ m+1c^* = g(x^*) - \sum_{j=1}^{n} g^* x_j^* \] (23)

and \( x^* \) is an arbitrary grid point. Hence the net price is given by:

\[ i^b - \left[ \sum_{j=1}^{n} f^* (p_j - x^*_j) + i^f(x^*) \right] i = 1 \ldots m \] (24)

or

\[ - \left[ - \sum_{j=1}^{n} g^* (p_j - x^*_j) - g(x^*) - p_{n+1} \right] \] (25)

Now the bracketed quantity in (24) or (25) is the ordinate of this particular tangent plane evaluated at the foot point \((p_1 \ldots p_n)\) in the case of (24) or \((p_1 \ldots p_n, -p_{n+1})\) for (25).

The definition of convexity (10) can be rewritten in the form:

\[ f(x) - f(x^*) > \frac{f[x^* + \gamma(x - x^*)] - f(x^*)}{\gamma} \] (26)

and letting \( \gamma \to 0 \) we have:

\[ f(x) \geq f(x^*) + \sum_{j=1}^{n} f^* (x - x^*_j) \] (27)
Replacing in (27) the argument $x$ by $p$ this inequality implies that the maximum value of $f(x^*) + \sum_{j=1}^{n} f_j^* (p - x^*)$ as a function of the tangential point $x^*$ but for fixed $p$ occurs for $x^* = p$ and that the value of this maximum is given by $f(p)$.

Returning to (24) and (25) we see that the bracketed quantities are maximum for $x^*_j = p_j$. Hence the algebraically smallest net price that can be computed from any selected column in the $i^{th}$ restriction set is given by:

$$i^b - i^f (p_1 \ldots p_n) \quad i = 1 \ldots m \quad (28)$$

or

$$g(p_1 \ldots p_n) + p_{n+1} \quad i = m+1 \quad (29)$$

Thus if either (29) or (28) for some $i$ is negative, the corresponding column vector is eligible to come into the basis. The components of the vector selected to enter the basis are then computed by evaluating the appropriate partial derivatives at the current foot point. Regard them as the $n+1$ elements of the "incoming column" and proceed with the modified simplex method.

We notice that the procedure selects the vector in the $i^{th}$ set with the smallest net price so that if this value is positive we proceed to the next (say) $i+1^{st}$ set. When the simplex procedure has been completed, the solution to the
problem is found above the dual slack in the tableau. Thus
the value of $x_{n+1}$ at optimum is given by $-p_{n+1}$ and the
depth of this optimum are given by $p_1 \ldots p_n$. The
question may be raised whether the foot point $p$ (at which the
partial derivatives are evaluated) is one of the grid points
and therefore whether the vector of computed partials is
actually one of the columns of the dual tableau. This ques-
tion is of no practical importance since we are at liberty to
choose the grid width to coincide with the highest decimal
accuracy that is achievable on the computer used.

The formal convergence of the dual simplex method is
insured by the finiteness of the tableau provided the
"incoming columns" are vectors of partial derivatives
(evaluated at $p$) computed for such surfaces for which the
net price given by (28) or (29) is negative. (In the case
of the slack vectors the net prices are, of course, obtained
by forming the inner product with the pricing vector.) If
the standard rule for simplex calculations was followed, one
would at each stage find the most negative net price from
among the $m+1$ values given by (28) and (29) and the $n+1$ slack
vectors in the tableau. It should be noted (and this is the
essential feature of the present method) that this would only
require the evaluation of $m+n+2$ net prices and not the
enormous number of net prices which would be involved if one
were evaluated for each column of Tableau 1.
In practice, however, some attention should be given to alternative modes of selecting the next incoming column from among the candidates yielding negative net prices. In order to preserve as far as possible the independence among the column vectors in the basis, we prefer the following procedure:

(i) If the point $p$ is in the cube $|p_j| \leq D$:
   (a) Compute (28) and (29) first for all those surfaces $f(x)$ and $-g(x)$ for which no tangential plane is already in the basis. Adjoin to these the $n+1$ dot products of the pricing vector with the slack vectors (i.e. the elements of the pricing vector itself) and select the most negative net price from this set.
   (b) If all net prices in (a) are positive or zero select the most negative net price from among the remaining surfaces. Note that the surfaces of the cube need not be tried since their net price is given by $D \pm p_j$ and will be positive.

(ii) If the point $p$ is outside the cube $|p_j| \leq D$, i.e. $|p_j| > D$ for at least one $j$, bring in an appropriate cube surface as the next vector. That is, bring in a corresponding unit vector with a $D$ in the objective row. In practice it will be found that with $D$ chosen sufficiently large, this situation will not arise. In fact, it is in general not necessary to decide on a value for $D$. 
Numerical examples

To illustrate the computational procedure let us consider the following examples:

Example 1: To find the maximum value of

$$g(x) = 25 - (x_1 - 10)^2 - (x_2 - 4)^2$$

subject to the restrictions

$$\frac{(x_1 - 5)^2}{36} + \frac{x_2^2}{16} \leq 1$$

$$\quad (x_1 - 6)^2 - x_2 \leq 2$$

$$x_1, x_2 \geq 0$$

The dual simplex problem is summarized in Tableau 2. The net price for each of the sets of restrictions, A, B, and C, are computed by substituting the current vector $\{p_1, \ldots, p_n, p_{n+1}\}$ into the expressions:

$$1 - \left[ \frac{(p_1 - 5)^2}{36} + \frac{p_2^2}{16} \right]$$

$$2 - \left[ (p_1 - 6)^2 - p_2 \right]$$

$$25 + p_3 - (p_1 - 10)^2 - (p_2 - 4)^2$$
Tableau 2. Summary of Example 1

<table>
<thead>
<tr>
<th>P_0</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>S_1</th>
<th>S_2</th>
<th>S_3</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 + \frac{x_1^2 - 25}{36} + \frac{x_2^2}{16}</td>
<td>x_1^2 - 34</td>
<td>x_1^2 + x_2^2 - 91</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>0</td>
<td>- \frac{(x_1 - 5)}{18}</td>
<td>-2(x_1 - 6)</td>
<td>-2(x_1 - 10)</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>- \frac{x_2}{8}</td>
<td>1</td>
<td>-2(x_2 - 4)</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
Table 1. Results for Example 1

<table>
<thead>
<tr>
<th>k</th>
<th>$p_1$</th>
<th>$p_2$</th>
<th>$-p_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>2.833333</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>2.008486</td>
<td>9.897991</td>
<td>25</td>
</tr>
<tr>
<td>4</td>
<td>4.262679</td>
<td>5.859122</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>6.268218</td>
<td>2.265812</td>
<td>25</td>
</tr>
<tr>
<td>6</td>
<td>7.200321</td>
<td>5.142314</td>
<td>25</td>
</tr>
<tr>
<td>7</td>
<td>7.824582</td>
<td>3.798986</td>
<td>25</td>
</tr>
<tr>
<td>8</td>
<td>9.193822</td>
<td>0.852482</td>
<td>25</td>
</tr>
<tr>
<td>9</td>
<td>8.766880</td>
<td>5.473178</td>
<td>25</td>
</tr>
<tr>
<td>10</td>
<td>8.490320</td>
<td>3.706743</td>
<td>23.086629</td>
</tr>
<tr>
<td>11</td>
<td>8.396695</td>
<td>3.735370</td>
<td>22.690763</td>
</tr>
<tr>
<td>12</td>
<td>8.320138</td>
<td>3.354005</td>
<td>22.204272</td>
</tr>
<tr>
<td>13</td>
<td>8.314380</td>
<td>3.356328</td>
<td>22.180182</td>
</tr>
<tr>
<td>14</td>
<td>8.314380</td>
<td>3.356328</td>
<td>21.744357</td>
</tr>
<tr>
<td>15</td>
<td>8.310157</td>
<td>3.336657</td>
<td>21.704280</td>
</tr>
<tr>
<td>16</td>
<td>8.310081</td>
<td>3.336274</td>
<td>21.703978</td>
</tr>
</tbody>
</table>

For this solution the vector $V$ corresponding to the tangent plane to $g(x)$ at $x_1 = 10$, $x_2 = 4$ was used with $S_1$ and $S_2$ to give the initial basis. The remainder of the solution proceeds just as in the modified simplex method with the
exception of the manner in which we select the next vector to enter the basis.

For a more detailed account of the procedure refer to Appendix B where the computations involved in this example are shown.

Table 1 lists the vectors \((p_1, p_2, -p_3)\) at the \(k^{th}\) stage of the iteration. Thus \(-p_3\) is the value of the "direct profit" at each step and \((p_1, p_2)\) are the coordinates at which it occurs. Formally, of course, one should continue until the point \(p\) is determined to within grid accuracy. In practice the process will be truncated at a point exhibiting certain convergence features such as the most negative net price being greater than \(-\varepsilon\) for some small value of \(\varepsilon\).

The solution to the problem correct to five places is \(x_1 = 8.31003, x_2 = 3.33624,\) and \(x_3 = 21.70342\). The net prices corresponding to (32), (33), and (34) after the last iteration are, respectively, \(-.000021, -.000200,\) and \(-.000336\).

Example 2: To find the maximum of

\[ g(x) = 25 - (x_1 - 5)^2 - x_2^2 \quad (35) \]

subject to the restrictions

\[ x_1^2 - x_2 \leq 0 \quad x_2 \leq 10 \quad x_1, x_2 \geq 0 \quad (36) \]

The dual simplex problem is summarized in Tableau 3.
Tableau 3. Summary of Example 2

<table>
<thead>
<tr>
<th>$P_0$</th>
<th>$D$</th>
<th>$E$</th>
<th>$F$</th>
<th>$S_1$</th>
<th>$S_2$</th>
<th>$S_2$</th>
<th>$V$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$x_1^2$</td>
<td>10</td>
<td>$x_1^2 + x_2^2$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>$-2x_1$</td>
<td>0</td>
<td>$-2(x_1 - 5)$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>$-1$</td>
<td>$-2x_2$</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-1</td>
</tr>
</tbody>
</table>

The net prices for the sets of restrictions $D$ and $F$ are computed from:

$$ p_2 - p_1^2 $$  \hspace{1cm} (37)

$$ 25 + p_3 - (p_1 - 5)^2 - p_2^2 $$  \hspace{1cm} (38)

The computation proceeds as in Example 1 with the vectors $S_1$, $S_2$, and $V$ forming the initial basis. In contrast to the first example, the solution to the present problem does not occur at a "vertex" of the restraining region and
Table 2. Results for Example 2

<table>
<thead>
<tr>
<th>k</th>
<th>$p_1$</th>
<th>$p_2$</th>
<th>$-p_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>2.500000</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>2.500000</td>
<td>6.250000</td>
<td>25</td>
</tr>
<tr>
<td>4</td>
<td>1.828711</td>
<td>2.893555</td>
<td>18.287025</td>
</tr>
<tr>
<td>5</td>
<td>1.469227</td>
<td>1.096114</td>
<td>14.692187</td>
</tr>
<tr>
<td>6</td>
<td>1.227486</td>
<td>-0.112580</td>
<td>12.274834</td>
</tr>
<tr>
<td>7</td>
<td>1.863770</td>
<td>3.068783</td>
<td>9.793677</td>
</tr>
<tr>
<td>8</td>
<td>1.477413</td>
<td>2.120281</td>
<td>9.144728</td>
</tr>
<tr>
<td>9</td>
<td>1.269494</td>
<td>1.609845</td>
<td>8.795496</td>
</tr>
<tr>
<td>10</td>
<td>1.126402</td>
<td>1.258556</td>
<td>8.755514</td>
</tr>
<tr>
<td>11</td>
<td>1.197947</td>
<td>1.434195</td>
<td>8.523440</td>
</tr>
<tr>
<td>12</td>
<td>1.233442</td>
<td>1.521338</td>
<td>8.507707</td>
</tr>
<tr>
<td>13</td>
<td>1.251470</td>
<td>1.565595</td>
<td>8.499717</td>
</tr>
<tr>
<td>14</td>
<td>1.242466</td>
<td>1.543493</td>
<td>8.499146</td>
</tr>
<tr>
<td>15</td>
<td>1.237904</td>
<td>1.532295</td>
<td>8.498857</td>
</tr>
<tr>
<td>16</td>
<td>1.237866</td>
<td>1.532311</td>
<td>8.498514</td>
</tr>
</tbody>
</table>

as we approach the true solution we have vectors in the basis which correspond to tangent planes which are nearly parallel. Thus the matrix of basis vectors is approaching singularity and we expect a loss of accuracy in the computations. From the fact that the trial solution occurs at the intersection of the relevant boundary planes at each stage (see next section) we might also expect the solution to "wander" rather violently because of this parallelism. As noted in Table 2 where the trial solutions are listed at each stage, this does
not appear to be too serious a problem. The solution to the problem correct to five places is \( x_1 = 1.23477, x_2 = 1.52466, x_3 = 8.49845 \). The net prices corresponding to (37) and (38) after the final iteration are \(-.000002\) and \(-.000143\).

Example 3: To find the maximum of

\[
g(x) = -x_1 + x_2
\]  

subject to the restrictions

\[
3x_1^2 - 2x_1 x_2 + x_2^2 \leq 1 \quad x_1, x_2 \geq 0
\]

As above, we summarize the problem in Tableau 4.

Tableau 4. Summary of Example 3

<table>
<thead>
<tr>
<th>( P_0 )</th>
<th>( G )</th>
<th>( H )</th>
<th>( S_1 )</th>
<th>( S_2 )</th>
<th>( S_2 )</th>
<th>( A )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 + 3x_1^2 + x_2^2 - 2x_1 x_2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>0 - (6x_1 - 2x_2)</td>
<td>-1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0 - 2(x_2 - x_1)</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>
The net price for the set of restrictions $G$ is given by

$$1 - [3p_1^2 - 2p_1 p_2 + p_2^2]$$ (41)

The other net prices are just the ordinary inner products. The initial basis consists of the vectors $S_1$, $S_2$ and the artificial vector $A$.

The primary reason for introducing this third example is that it is solved by the cutting plane method in Kelley [21] and thus affords us a comparison of the two methods. The correct solution is seen to be $x_1 = 0$, $x_2 = 1.0$, $x_3 = 1.0$, and the cutting plane method after nine iterations yields $x_1 = -.07348$, $x_2 = .92972$, $x_3 = 1.00321$. In Table 3, we see that the present method does much better in five iterations with considerably less work as we require only one "simplex iteration" at each stage rather than the complete solution of a linear program.

Table 3. Results for Example 3

<table>
<thead>
<tr>
<th>$k$</th>
<th>$p_1$</th>
<th>$p_2$</th>
<th>$-p_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1.250000</td>
<td>1.250000</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1.025000</td>
<td>1.025000</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1.000305</td>
<td>1.000305</td>
</tr>
</tbody>
</table>
In fairness to the cutting plane method, it should be noted that the boundary plane $x_1 = 0$ is in our initial basis and this probably accounts for some of the acceleration in the convergence.

**Relations between the direct and dual problems**

For convenience we restate the direct and dual problems considered above in a slightly abbreviated form. The dual problem given by (18) and (19) is written as:

$$\max - w' c$$

subject to

$$-A' w \leq -d$$

$$w \geq 0$$

and hence the direct problem can be written as

$$\max x_{n+1} = d' x$$

subject to

$$A x \leq c$$

$$x \geq 0$$

Note here that the vector $x$ is the column vector $(x_1, x_2 \ldots x_{n+1})$. Consider the situation at any stage of the dual problem. Let $\hat{B}$ be the matrix of basis vectors at
this stage and let $-\hat{c}'$ be the vector of values corresponding to these basis vectors. Then the part of the pricing vector given by $(p_1 \ldots p_{n+1})$ is expressed simply as $-\hat{c}' \hat{B}^{-1}$ [see for example Charnes and Cooper, 7, p. 472]. The column vectors in the dual problem correspond to linear restrictions in the direct problem and hence the set of bounding planes which are relevant at any stage of the problem can be expressed by the following system of equations:

$$-\hat{B}' G x = \hat{c}$$

(46)

Here $G$ is an identity matrix with the sign changed in the last row. Thus $-G$ takes care of the sign changes which have been made in setting up the dual tableau. The foot point $(p_1 \ldots p_n, -p_{n+1})$ used in the algorithm to determine the next incoming column can now be written as:

$$-\hat{c}' \hat{B}^{-1} G$$

(47)

Thus since

$$-\hat{B}' G G' (\hat{B}')^{-1} (-\hat{c}) = \hat{c}$$

(48)

it follows that the foot point in the $n+1$ dimensional $x$-space lies at the intersection of the current set of bounding planes.

The "direct profit" at any stage is given by $x_{n+1} = d' x$ and we now show that the sequence of $x_{n+1}$ values generated as
we proceed with the solution of the dual problem is monotone non-increasing.

The "dual profit" at the $i^{th}$ stage is given by $-c' \hat{w}$ where $\hat{w} = B^{-1} d$. The simplex method then selects a new vector to come into the basis in such a way that if $-c' \tilde{w}$ is the profit at the next stage, then,

$$-c' \tilde{w} \geq -c' \hat{w} \quad (49)$$

Now the direct profit at the $i^{th}$ stage is given by:

$$\hat{x}_{n+1} = d' x = d' G (B')^{-1} (-\hat{c}) = (B^{-1} d)' \hat{c} = \hat{w}' \hat{c} \quad (50)$$

and similarly at the next stage $\tilde{x}_{n+1} = \tilde{w}' \tilde{c}$. Thus we have

$$\hat{x}_{n+1} = \hat{w}' \hat{c} \geq \tilde{w}' \tilde{c} = \tilde{x}_{n+1} \quad (51)$$

Extensions of the method

Computation of partial derivatives After the algorithm has selected the set of restrictions from which the next vector is to be chosen, it is necessary to compute the components of this vector by evaluating the appropriate partial derivatives at the current foot point. In practice, this procedure may be undesirable either because (i) we may not wish to have a subroutine for each of the derived
functions or (ii) the functions may not be described analytically but only given by ordinates at certain points. In either case, we may use an approximation to the exact partial derivatives. For example, the partial derivative of $f(x_1 \ldots x_p)$ with respect to $x_1$, evaluated at the point $\hat{x}$ may be approximated by

$$\frac{f(\hat{x}_1 + \varepsilon, \hat{x}_2 \ldots \hat{x}_p) - f(\hat{x}_1, \hat{x}_2 \ldots \hat{x}_p)}{\varepsilon}$$

(52)

for some small $\varepsilon$. Of course, more elaborate approximations may be used depending on the nature of the functions [see, e.g., Hildebrand, 17].

**Non-convex functions** The algorithm presented here depends on the convexity of the objective function (if we think of the problem as a minimization) and of the constraint functions. If, for example, the objective function is convex over part of its domain but concave in another part, then the algorithm obviously fails since the region defined by the tangent planes will no longer contain the true feasible region. In certain special cases we may know in advance the subregions over which the objective function is convex. Then an algorithm which finds the minimum in each of these subregions (possibly by redefining the functional outside of the region) and then selects the smallest of these would seem reasonable.
In still other situations the objective function may be convex but the feasible region may be non-convex. In this case it may be possible to divide the feasible region into several convex subregions and find the minimum over each of these. For example, we might devise a method in which we subdivide the restraining region by planes into convex, or nearly convex, subregions and thereby determine the approximate location of the optimum. Then, by investigating this area more thoroughly we can obtain the correct location of the optimum.

In general it is clear that without considerable a priori knowledge of the nature of the functional and the restraining functions it will be difficult to extend the present algorithm to the non-convex situation.
PART II. THE APPLICATION OF MATHEMATICAL PROGRAMMING TO STATISTICAL ESTIMATION THEORY
REVIEW OF LITERATURE

The problem of estimating the parameters in a linear regression model is generally solved by the classical method of least squares. This method requires the minimization of a quadratic function of the parameters, referred to as the residual sum of squares. If we attempt to use the method of least squares when it is known that the parameters are constrained so as to lie in a subspace of the parameter space then we are faced with a quadratic programming problem. That is, the problem of minimizing a quadratic form subject to certain inequality constraints. If the constraints are linear there are many solutions to the problem. We mention here the methods of Beale [5], Hildreth [18], Markowitz [24], and Wolfe [32].

The particular quadratic form which arises in the method of least squares is known [see, e.g., Graybill, 15] to be at least positive semidefinite. This is an essential requirement of quadratic programming methods since the function to be minimized can then be shown to be convex. By a linear transformation of the variables we can reduce such a quadratic form to one which is a "separable function" of the new variables. (The term, "separable function", as defined by Charnes and Lemke [10] refers to a function which is a sum of functions each depending on only one of the variables.) Thus the
method of Charnes and Lemke [10] for minimizing a convex separable function can be applied to the constrained regression problem if the constraints are linear. Hartley [16] presents an algorithm for solving this problem for special types of nonlinear restraints, i.e., separable functions with certain convexity restrictions.

There are, of course, alternate criterion for fitting a linear model such as the methods of least absolute deviations and least maximum deviation. These methods do not enjoy the statistical properties of least squares, but have certain computational advantages and in some cases are conceptually more appropriate.

Wagner [30] discusses both of these approaches to the linear regression problem. In particular, he considers the two problems: (i) find the regression coefficients $b_j$ which minimize $\sum_i |\sum_j x_{ij} b_j - y_i|$ and (ii) find the coefficients $b_j$ which minimize $\max_i |\sum_j x_{ij} b_j - y_i|$. By employing the dual theorem of linear programming he shows how problem (i) may be solved as a linear programming problem with bounded variables involving a number of restrictions equal to the number of regression coefficients, say $p$. Problem (ii) is reduced to an ordinary linear programming problem with $p+1$ restrictions. Similar approaches to the regression problem are taken by Karst [19], Kelley [20]; and Charnes et al. [9],
who also suggest some statistical properties of these estimates.

Charnes and Cooper [7, 8] generalize these notions to a much broader class of nonlinear problems. Specifically, they consider the minimization of the functional

$$F(\lambda_1 \ldots \lambda_n) = \sum_k w_k \left| \sum_r a_{rk} \lambda_r - s_k \right| \text{ subject to linear restrictions on the } \lambda's.$$ 

Little attention has been given to problem of fitting a linear model subject to nonlinear restrictions by the method of least squares except that computationally it is a special case of convex programming. In what follows, we propose to develop some of the statistical theory resulting from the application of the method of least squares to this problem as well as suggest means of computing the estimates.
THE THEORY OF LEAST SQUARES ESTIMATION
WITH CONSTRAINED PARAMETERS

Preliminary remarks

The problem with which we are concerned is that of fitting a linear model when the parameters are constrained so as to lie in a convex region in the parameter space. In terms of the usual notation in linear regression [see Graybill, 15] we are looking for a vector, say \( \tilde{\beta} \), which gives the best fit in the least squares sense to the linear model

\[
y = X \beta + e
\]

subject to the restriction that \( \tilde{\beta} \) lie in the convex region \( S_\beta \). Here \( y \) is an \( n \times 1 \) vector of observed values, \( X \) is an \( n \times p \) matrix of fixed known constants and \( \beta \) is a \( p \times 1 \) vector of unknown parameters. The \( n \times 1 \) vector \( e \) denoting the difference between the observations \( y \) and their expected values \( X\beta \) is assumed to be normally distributed with mean zero and covariance matrix \( \sigma^2 I_n \) where \( \sigma^2 \) is unknown and \( I_n \) is an \( n \times n \) identity matrix.

The classical unrestrained least squares solution requires the minimization of the residual sum of squares, i.e.,

\[
Q = e' e = (y - X\beta)' (y - X\beta)
\]

The vector \( \hat{\beta} \) which minimizes the residual sum of squares
satisfies the normal equations

\[ X' X \beta = X' y \]  

(55)

If we assume that the rank of the \( X \) matrix is \( p \) then the inverse of the \( X' X \) matrix exists and the vector \( \hat{\beta} \) is unique. In what follows, we shall assume that we are in this full rank situation since we can always express the non-full rank model as a full rank model by a linear transformation [see Graybill, 15, p. 235].

This \( \hat{\beta} \) estimator has many desirable properties, a few of which we recall here. It is easily shown that \( \hat{\beta} \) is an unbiased estimate of \( \beta \) with covariance matrix \( \sigma^2 S^{-1} \), where \( S = X' X \). It is the best linear unbiased estimator (b.l.u.e.) in the sense that any other linear estimate \( \beta^* \) will have a greater variance for estimating the individual \( \beta_i \). We recall also that

\[ S^2 = \frac{(y - X \hat{\beta})' (y - X \hat{\beta})}{n - p} \]  

(56)

is an unbiased estimate of \( \sigma^2 \).

Rewriting (54) as

\[ Q(\beta) = y' y - 2y' X \beta + \beta' (X' X) \beta \]  

(57)

we see that \( Q(\beta) \) contains a quadratic form \( \beta' S \beta \) which is
easily shown to be positive definite and hence \( Q(\beta) \) is a convex function [see Kuhn and Tucker, 23].

By the restriction of convexity on the region \( S_\beta \) we mean simply that if \( \beta_1 \) and \( \beta_2 \) are any two points in \( S_\beta \) then any point on the line segment joining \( \beta_1 \) and \( \beta_2 \) also lies in \( S_\beta \) [see, e.g., Bonnesen and Fenchel, 6]. We note that any such point may be expressed as

\[
\beta_3 = \theta \beta_1 + (1 - \theta)\beta_2 \quad 0 \leq \theta \leq 1
\]  

(58)

If \( f(\beta) \) is a convex function as defined by (10) then we see that the set of points which satisfy \( f(\beta) \leq c \) is a convex set, for if \( \beta_1 \) and \( \beta_2 \) are in this set then by (10) we see that

\[
f(\beta_3) = f[\theta \beta_1 + (1 - \theta)\beta_2] \\ \leq \theta f(\beta_1) + (1 - \theta)f(\beta_2) = c
\]

(59)

and hence \( \beta_3 \) is in the set. The region \( S_\beta \) will usually be defined by the intersection of several convex regions each of which is defined by an inequality of the type

\[
f(\beta) \leq c
\]

(60)

where \( f(\beta) \) is a convex function.

Thus we see that the problem of determining \( \tilde{\beta} \) falls into the class of problems for which the convex programming procedure in Part I was devised, and the numerical determination
of $\hat{\beta}$ is thus achieved.

In general, the restriction that $S_\beta$ be a convex region is not quite equivalent to the condition that it be defined by inequalities of the type (60) for convex functions. There is, in fact, a slightly larger class of functions, called quasi-convex functions, which may define a convex region $S_\beta$. Quasi-convex functions are defined by having the property that the set of all $\beta$ for which $f(\beta) \leq c$ is a convex set. That is to say, if $f(\beta) \leq f(\hat{\beta})$ then

$$f [\theta \beta + (1 - \theta) \hat{\beta}] \leq f(\hat{\beta}) \quad 0 \leq \theta \leq 1 \quad (61)$$

Clearly, every convex function is quasi-convex, but the converse is not true. In many cases, we may express the region defined by a quasi-convex function in terms of a convex function. For example, the function $\log (x^2 + y^2)$ is quasi-convex but the region $\log (x^2 + y^2) \leq c$ can just as well be described by the inequality $x^2 + y^2 \leq e^c$ and $x^2 + y^2$ is a convex function. The situation is not always this simple, however, as seen by looking at further examples. We give as additional examples the functions $x_1 x_2$ and

$$[(x - 1) + [(1 - x)^2 + 4(x + y))]^{1/2}$$

which are quasi-concave (i.e., their negatives are quasi-convex). We refer to the work of Arrow and Enthoven [3] for a discussion of these functions and an extension of the Kuhn-Tucker [23] saddle-point theory to the case of quasi-convex functions.
We shall develop, in a later section, a special quadratic programming algorithm which depends only on the convexity of the region $S$ and allows the defining functions to be quasi-convex.

The canonical form and determination of $\beta$

Since the symmetric $p \times p$ matrix $S = X'X$ is positive definite there exists a non-singular matrix $A$ such that

$$A' SA = I_p$$

(62)

(The recurrence relations defining a triangular matrix with this property are given in Faddeeva [13, p. 81].) If we consider the linear transformation

$$\beta = A \gamma$$

(63)

and let $U = XA$ we may rewrite the linear model (53) in terms of new regression coefficients $\gamma$ as follows:

$$\gamma = XA A^{-1} \beta + e = U \gamma + e$$

(64)

By definition,

$$U'U = A' X' XA = A' SA = I_p$$

(65)

hence the b.l.u.e. of $\gamma$ is, from the normal equations,

$$\hat{\gamma} = U' \gamma$$

(66)
and the covariance matrix of \( \hat{\gamma} \) is \( \sigma^2 I_p \). Transforming back to the original parameter space we see that

\[
\hat{\beta} = A \hat{\gamma}
\]

(67)

and it is easily seen that \( \hat{\beta} \) does satisfy the normal equations given by (55). The minimization of (54) is thus equivalent to the minimization of

\[
H(\gamma) = (\gamma - U\gamma)' (\gamma - U\gamma)
\]

(68)

which we may write in terms of the estimate \( \hat{\gamma} \) as

\[
H(\gamma) = (\gamma - U\hat{\gamma})' (\gamma - U\hat{\gamma}) + (\hat{\gamma} - \gamma)' (\hat{\gamma} - \gamma)
\]

(69)

We note that the last term of equation (69) is the squared distance from \( \hat{\gamma} \) to any point \( \gamma \) in the parameter space. (For brevity we write this term as \( |\hat{\gamma} - \gamma|^2 \).) It follows that the contours of equal \( H \) are hyperspheres about the least squares point \( \hat{\gamma} \).

Let the image of \( S_\beta \) under the linear transformation \( \gamma = A^{-1} \beta \) be expressed as \( A^{-1} S_\beta = S_\gamma \). The property of convexity is seen to be invariant under this transformation for if \( \gamma_1 \) and \( \gamma_2 \) are points in \( S_\gamma \), say the images of \( \beta_1 \) and \( \beta_2 \) in \( S_\beta \), then any convex linear combination (i.e. \( \theta \gamma_1 + (1 - \theta) \gamma_2, 0 \leq \theta \leq 1 \)) is in \( S_\gamma \). That is to say,

\[
\theta \gamma_1 + (1 - \theta) \gamma_2 = A^{-1}(\theta \beta_1 + (1 - \theta) \beta_2) = A^{-1} \beta_3
\]

(70)
and by the convexity of $S_{\beta}$, $\beta_3$ is in $S_{\beta}$ and hence $A^{-1}\beta_3$ is in $S_{\gamma}$.

Now by equation (69) we see that the minimization of $H(\gamma)$ subject to the restriction that the solution must lie in the convex region $S_{\gamma}$ reduces to finding that point $\gamma$ in $S_{\gamma}$ which minimizes $|\gamma - \gamma|^2$. In other words, we have the following solution:

$$
\begin{align*}
\gamma &= \left\{ \begin{array}{ll}
(i) & \gamma \text{ if } \gamma \text{ is in } S_{\gamma} \\
(ii) & \text{the orthogonal projection of } \\
& \gamma \text{ to } S_{\gamma} \text{ if } \gamma \text{ is not in } S_{\gamma}
\end{array} \right. \\
&= (71)
\end{align*}
$$

If we now transform back to the $\beta$-space to determine $\tilde{\beta} = A \tilde{\gamma}$ we see that $\tilde{\beta}$ does minimize $Q(\beta)$ subject to the condition that $\tilde{\beta}$ be in $S_{\beta}$, for if not, suppose $Q(\beta_*) < Q(\tilde{\beta})$ for some $\beta_*$ in $S_{\beta}$. Then $\gamma_* = A^{-1}\beta_*$ is in $S_{\gamma}$ and $H(\gamma_*) = Q(\beta_*) < Q(\tilde{\beta}) = H(\tilde{\gamma})$ which is a contradiction.

In summary, we see that the vector $\tilde{\beta}$ in $S_{\beta}$ which minimizes the residual sum of squares, $Q(\beta)$, can be determined by transforming to the $\gamma$-space, projecting the least squares point $\hat{\gamma}$ onto $S_{\gamma}$ to determine $\gamma$ and then computing $\tilde{\beta} = A \gamma$.

**Properties of $\gamma$**

We note here some simple properties of the $\gamma$ estimator which are a consequence of the geometry. We shall assume that
the true parameter point, say \( \gamma_0 \), does lie in \( S_\gamma \). If the region \( S_\gamma \) is accurately specified then, of course, \( \gamma_0 \) will lie in \( S_\gamma \). In practice, a simple test in a probabilistic sense is given in the following:

Let \( R \) be the radius of a sphere about the least squares point \( \hat{\gamma} \) such that

\[
\Pr \left[ |\hat{\gamma} - \gamma_0|^2 \leq R^2 \mid \gamma_0 \right] = 1 - \alpha \tag{72}
\]

If the intersection of this sphere with the region \( S_\gamma \) is empty we reject the assumption that \( \gamma_0 \) is in \( S_\gamma \) with \( (1 - \alpha) \) per cent confidence. From the usual considerations in linear hypothesis theory [see, e.g., Graybill, 15] we know that

\[
\frac{|\hat{\gamma} - \gamma_0|^2}{ps^2} \tag{73}
\]

is distributed as \( F \) with \( p \) and \( n-p \) degrees of freedom. (Here \( s^2 \) is the unbiased estimate of \( \sigma^2 \).) It follows that

\[
\Pr \left[ |\hat{\gamma} - \gamma_0|^2 \leq ps^2 F_{p, n-p} (1 - \alpha) \right] = 1 - \alpha \tag{74}
\]

and hence we compute the value for \( R^2 \) in equation (72) as

\[
R^2 = ps^2 F_{p, n-p} (1 - \alpha). \tag{75}
\]

Comparing this with the value of \( |\hat{\gamma} - \gamma|^2 \) we see that if \( R^2 < |\hat{\gamma} - \gamma|^2 \), we reject the assumption that \( \gamma_0 \) is in \( S_\gamma \).

As a result of the convexity of \( S_\gamma \), it is easily shown
that if \( \gamma_0 \) is in \( S_\gamma \) then the projected estimator \( \gamma \) is closer
to \( \gamma_0 \) than is the least squares estimator \( \hat{\gamma} \). That is to say,
\[ |\gamma - \gamma_0| \leq |\hat{\gamma} - \gamma_0|. \]

The \((1 - \alpha)\) per cent confidence region for \( \gamma_0 \) is the
intersection of the sphere defined by \(|\gamma - \gamma|^2 \leq R^2\) and the
restraining region \( S_\gamma \), where \( R^2 \) is determined as above.
Confidence limits on \( \gamma_0 \) may be determined by finding the maximum and minimum of each coordinate in the intersection
region. That is, we solve for \( i = 1 \ldots p \) the two convex
programming problems

\[
\begin{align*}
\text{maximize } & \gamma_i \\
\text{and } & \\
\text{minimize } & \gamma_i
\end{align*}
\]  

subject to the restrictions

\[
\gamma \text{ in } S_\gamma
\]  

and

\[
|\gamma - \hat{\gamma}|^2 \leq R^2
\]  

If the region \( S_\gamma \) is defined by convex functions, then since
\(|\gamma - \hat{\gamma}|^2\) is also a convex function we may use the convex
programming algorithm developed in Part I to solve problems
(75) and (76) with restraints (77). In view of the magnitude
of the computations required to determine these confidence
limits we may choose to base our confidence limits on the
asymptotic moments developed in the next section, or on approximate moments based on the exact distribution derived in the next chapter.

**Asymptotic properties of \( \tilde{\gamma} \) and \( \tilde{\beta} \)**

Suppose we have \( N \) replicates of the vector \( \gamma \) for a given \( U \) matrix in the canonical model given by equation (64). Then we may base the least squares estimation on the model

\[
\bar{\gamma} = U \gamma + \bar{\varepsilon}
\]

(78)

where \( \bar{\gamma} \) is the vector of means and the vector \( \bar{\varepsilon} \) is assumed to be normally distributed with mean zero and covariance matrix \((\sigma^2/N)I_n\). Using this model we now investigate some of the asymptotic properties of the estimator \( \tilde{\gamma} \) and hence also of \( \tilde{\beta} \).

**Consistency** By virtue of the consistency of \( \hat{\gamma} \) and the fact that \(|\tilde{\gamma} - \gamma_0| \leq |\hat{\gamma} - \gamma_0|\), it follows that \( \tilde{\gamma} \) is consistent.

**Asymptotic normal behavior of \( \tilde{\gamma} \)** We consider separately, case (i) in which \( \gamma_0 \) is an interior point of \( S_{\gamma} \) and case (ii) in which \( \gamma_0 \) is a regular boundary point (see Appendix C) of \( S_{\gamma} \).

In case (i), let \( R \) be the radius of the largest sphere about \( \gamma_0 \) which is contained in \( S_{\gamma} \). Then, since \(|\tilde{\gamma} - \gamma_0|^2 \) is distributed as \( \chi^2_p \sigma^2/N \) we have:
\[ \Pr [\hat{\gamma} \text{ is in } S_\gamma] \geq \Pr [|\hat{\gamma} - \gamma_0|^2 \leq R^2] \]
\[ = \Pr [\chi_p^2 \sigma^2/N \leq R^2] = \Pr [\chi_p^2 \leq NR^2/\sigma^2] \quad (79) \]

The expression in (79) tends to 1 as \( N \) is increased. Thus \( \Pr[\tilde{\gamma} = \hat{\gamma}] \) tends to 1 as \( N \) tends to \( \infty \), and the distributions of \( \hat{\gamma} \) and \( \tilde{\gamma} \) tend to be equal in probability as \( N \) gets large.

The distribution of \( \hat{\gamma} \) in case (i) is thus seen to be asymptotically normal.

This is not true in case (ii) as can be seen by considering the following linear transformation:

\[ \eta = G \gamma \quad (80) \]

Here \( G \) is an orthogonal matrix which effects a rotation of the \( \gamma \)-space so that the \( \eta_1 \) direction is normal to the tangent plane to \( S_\gamma \) at the point \( \gamma_0 \). Now in view of the fact that \( \hat{\gamma} \) is normally distributed about a mean \( \gamma_0 \) and with covariance matrix \( (\sigma^2/N)I_p \), it follows [see Anderson, p. 19] that \( \hat{\eta} = G \hat{\gamma} \) is also normally distributed with mean \( \eta_0 = G \gamma_0 \) and covariance matrix \( (\sigma^2/N)G G' = (\sigma^2/N)I_p \). Asymptotically, the joint distribution of \( \tilde{\eta} = G \tilde{\gamma} \) is then given by

\[ (2\pi \sigma^2/N)^{-\frac{1}{2}} \exp \left[ -\frac{N}{2\sigma^2} \sum_{i=2}^{P} (\tilde{\eta}_i - \eta_{0i})^2 \right] \Phi(\tilde{\eta}_1) \quad (81) \]

where
\[
(2\pi \sigma^2/N)^{-1/2} \exp\left[-\frac{N}{2\sigma^2}(\tilde{\eta}_1 - \eta_0)^2\right] \text{ if } \tilde{\eta}_1 < \eta_0
\]
\[
\phi(\tilde{\eta}_1) = \begin{cases} 
1/2 & \text{if } \tilde{\eta}_1 = \eta_0, \\
0 & \text{if } \tilde{\eta}_1 > \eta_0
\end{cases}
\]

Asymptotic moments of \(\tilde{\gamma}\) and \(\tilde{\beta}\)

If \(\gamma_0\) is an interior point of \(S_\gamma\) as in case (i) then we have asymptotically the normal moments. That is to say, as \(N\) gets large, \(\tilde{\gamma}\) has expectation \(\gamma_0\) and

\[
E\left[(\tilde{\gamma} - \gamma_0)(\tilde{\gamma} - \gamma_0)'ight] = \left(\sigma^2/N\right)I_p 
\] 

(83)

and

\[
E\left[(\tilde{\beta} - \beta_0)(\tilde{\beta} - \beta_0)'ight] = \left(\sigma^2/N\right)S^{-1} 
\]

(84)

where \(E\) is the usual expectation operator.

In case (ii) where \(\gamma_0\) is a regular boundary point the asymptotic moments are given by

\[
E(\tilde{\eta}_i) = \eta_{0i} 
\]

(85)

\[
\text{Var}(\tilde{\eta}_i) = E(\tilde{\eta}_i - \eta_{0i})^2 = \sigma^2/N 
\]

for \(i = 2, 3, \ldots p\). For \(i = 1\) we have, in view of the joint distribution given by (81),
\[ E(\tilde{\eta}_1) = \frac{1}{2} \eta_01 + \int_{-\infty}^{\eta_{01}} \tilde{\eta}_1 \left( \frac{N}{2\pi \sigma^2} \right)^{1/2} \exp \left\{ -\frac{N}{2\sigma^2} (\tilde{\eta}_1 - \eta_01)^2 \right\} d\tilde{\eta}_1 \]

\[ = \eta_01 - \left( \frac{\sigma^2}{2\pi N} \right)^{1/2} \]

which tends to \( \eta_{01} \) as \( N \) tends to \( \infty \). The mean square error (M.S.E.) for \( \tilde{\eta}_1 \) is given by

\[ E(\tilde{\eta}_1 - \eta_{01})^2 = \int_{-\infty}^{\eta_{01}} (\tilde{\eta}_1 - \eta_{01})^2 \left( \frac{N}{2\pi \sigma^2} \right)^{1/2} \exp \left\{ -\frac{N}{2\sigma^2} (\tilde{\eta}_1 - \eta_{01})^2 \right\} d\tilde{\eta}_1 \]

\[ = \sigma^2/2N \]

and hence

\[ \text{Var} (\tilde{\eta}_1) = \text{M.S.E.} (\tilde{\eta}_1) - (\text{Bias in } \tilde{\eta}_1)^2 \]

\[ = \frac{\sigma^2}{2N} - \frac{\sigma^2}{2\pi N} \]

\[ = \frac{\sigma^2}{N} - \frac{\sigma^2}{2N} (1 + 1/\pi) \]

The covariance matrix of \( \tilde{\eta} \) is thus given by the partitioned matrix.
To relate these properties to the original $\beta$ variates in case (ii) we recall that the transformation from the $\beta$-space to the $\eta$-space is effected by the two transformations $\beta = A\gamma$ and $\eta = G\gamma$. We combine these in the single transformation $\beta = AG'\eta$, noting that $G^{-1} = G'$ by the orthogonality of $G$. If we let $T = AG'$, then the bias in $\tilde{\beta}$ is given by

$$E(\tilde{\beta} - \beta_0) = E[T(\tilde{\eta} - \eta_0)] = T(Bias in \tilde{\eta})$$

$$= T \begin{bmatrix} -\left(\frac{\sigma^2}{2\pi N}\right)^{1/2} \\ 0 \\ \vdots \\ 0 \end{bmatrix} = -\left(\frac{\sigma^2}{2\pi N}\right)^{1/2} T_1$$

(90)

Here $T_1$ is used to denote the first column of the $T$ matrix. The covariance matrix for $\tilde{\beta}$ is thus given by
\[ E[\tilde{\beta} - E(\tilde{\beta})][\tilde{\beta} - E(\tilde{\beta})]' = E[T[\tilde{\eta} - E(\tilde{\eta})][\tilde{\eta} - E(\tilde{\eta})]'T'] \]

\[ = \sigma^2/N T \begin{bmatrix} 1 - 1/2 (1 + 1/\pi) & 0 & ... & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & I_p-1 \end{bmatrix} T' \quad (91) \]

\[ = \sigma^2/N T T' - \sigma^2/2N (1 + 1/\pi) T_1 T_1' \]

Now by definition of \( T \) it follows that

\[ T T' = (A G') (A G')' = A G' G A' \]

\[ = A A' = S^{-1} \quad (92) \]

Further, we see that the matrix \( T_1 T_1' \) has as the element in the \( i \)th row and \( j \)th column the product \( t_{i1} t_{j1} \). In particular, the diagonal elements of \( T_1 T_1' \) are given by \( t_{i1}^2 \) and hence are positive. From this fact and equations (91) and (92) we see that when we can infer that \( \beta_0 \) is a regular boundary point of \( S_\beta \) we have a reduction in variance as compared with the variance of \( \hat{\beta}_1 \). That is, asymptotically

\[ \text{Var} (\tilde{\beta}_1) = \text{Var} (\hat{\beta}_1) - (\sigma^2/2N) (1 + 1/\pi) t_{i1}^2 \quad (93) \]

Similarly we see that
and thus the M.S.E. of \( \tilde{\beta}_i \) is the variance of \( \tilde{\beta}_i \) reduced by the amount \( t_{11}^2 \sigma^2/2N \).

Asymptotic variance estimation

Since estimation of variances and covariances is an a posteriori operation it is not appropriate to disregard information about the true parameter \( \gamma_0 \) which is contained in \( \hat{\gamma} \) or \( \tilde{\gamma} \). Inferences about \( \gamma_0 \) may be based on whether or not a regular boundary point of \( S_{\gamma} \) is in the confidence sphere about \( \hat{\gamma} \). We have the following criterion for estimating variances:

(i) If \( \hat{\gamma} \) permits the inference that \( \gamma_0 \) is an interior point of \( S_{\gamma} \), then we estimate \( \sigma^2/N \) by

\[
S^2/N = (\tilde{y}' \tilde{y} - \hat{\gamma}' \hat{\gamma})/n - p
\]

and use equations (83) or (84) for estimating variances and covariances.

(ii) If \( \hat{\gamma} \) permits the inference that \( \gamma_0 \) is a regular boundary point of \( S_{\gamma} \), we again estimate \( \sigma^2/N \) by (95) and use equation (91) or its obvious analogue in terms of \( \tilde{\gamma} \). The problem of approximating the matrix \( T_1 T_1' \) may be solved in the following manner. We write \( T_1 T_1' \) as

\[
T_1 T_1' = A \vee v' A'
\]
where $v'$ is the first row of $G$. By virtue of the fact that $G$ is an orthogonal matrix which rotates the axis of the $y$-space so that the $\eta_1$ direction is normal to $S_y$ at $y_0$, it follows that $v$ is a vector of unit length which is normal to $S_y$ at $y_0$. If $y$ is outside of $S_y$ then we may approximate $v$ by the vector

$$\frac{\tilde{y} - \hat{y}}{|\tilde{y} - \hat{y}|} \quad (97)$$

Thus the reduction in estimated variance may be readily obtained by using (95) and (97) with equation (93). If $\hat{y}$ is inside $S_y$ then there is not a convenient way to approximate the vector $v$. In fact, if we project $\hat{y}$ to the surface to determine a pseudo $\tilde{y}$ for use in expression (97) we find that this point is not in general unique. In this situation we might just ignore the reduction in variance available in expression (93) and use the normal moments. In any case the estimation of variance given here is quite crude and the formulas should be used with some reservations.
Preliminary remarks

In the previous section we have developed asymptotic expressions for the second moments of the projected least squares estimator $\tilde{\gamma}$. We now propose to develop the exact distribution of $\tilde{\gamma}$ for certain types of restraining regions, $S_{\gamma}$. We assume the canonical model $\gamma = U \gamma + \epsilon$ as given in equation (64) and recall that the vector $\epsilon$ is multivariate normal with mean zero and covariance matrix $\sigma^2 I_n$. The least squares solution $\tilde{\gamma}$ is then normally distributed with mean, say $\gamma_0$, and covariance matrix $\sigma^2 I_p$.

Let us assume that the convex restraining region $S_{\gamma}$ is bounded by a single surface, say

$$ F(\gamma) = 0 	ag{98} $$

We assume, further, that this surface has a unique tangent plane at every point on the surface, and that $F(\gamma) < 0$ for $\gamma$ inside $S_{\gamma}$.

Assuming, as above, that $\gamma_0$ is the true parameter point, we consider the translation of the $\gamma$-space given by

$$ x = \gamma - \gamma_0 \tag{99} $$

The joint distribution of the $\hat{x}_i$, $i = 1 \ldots p$, is then given
by

\[
(2\pi\sigma^2)^{-p/2} \exp \left[ -\frac{1}{2\sigma^2} \sum_{i=1}^{p} \hat{x}_i^2 \right]
\]  

(100)

It is clear that the distribution of the vector \( \hat{x} \), where \( \hat{x} = \hat{y} - Y_0 \), consists of the spherical normal distribution in the interior of the region \( S_\hat{y} \) and a distribution on the surface of \( S_\hat{y} \) as a result of the orthogonal projection of \( \hat{y} \) to \( S_\hat{y} \). To determine this surface distribution it is necessary to sum the probability along the directions normal to \( S_\hat{y} \).

For this purpose we assume that the surface defined by (98) is given parametrically in terms of the surface coordinates \( v_2 \ldots v_p \) by the equations (see Appendix C):

\[
x_i = i_f(v_2 \ldots v_p) \quad i = 1 \ldots p
\]  

(101)

when the parameters \( v_2 \ldots v_p \) vary over some region of the \( p-1 \) dimensional space of \( v_2 \ldots v_p \). Let us denote \( \frac{\partial f}{\partial v_j} \) by \( i_{fj} \) and consider the \( p \times p-1 \) matrix of these partial derivatives,

\[
M = \begin{bmatrix}
1^{i_f2} & 1^{i_f3} & \ldots & 1^{i_fp} \\
2^{i_f2} & 2^{i_f3} & \ldots & 2^{i_fp} \\
\vdots & \vdots & \ddots & \vdots \\
p^{i_f2} & p^{i_f3} & \ldots & p^{i_fp}
\end{bmatrix}
\]  

(102)
We let \( \tau_i, i = 1 \ldots p \), be the determinant of the \( p-1 \times p-1 \) matrix obtained by deleting the \( i \)th row of the matrix \( M \). It is shown in Appendix C that the vector

\[
\tau = (\tau_1, \tau_2 \ldots (-1)^{i-1}\tau_i \ldots (-1)^{p-1}\tau_p) \tag{103}
\]

evaluated at some surface point, is a normal vector to the surface at that point. The assumption that every point on the surface has a unique tangent plane implies that for no surface point do all of the \( \tau_i \) vanish.

It is convenient to consider a unit vector to the surface and also to remove the cumbersome negative signs in equation (103). For this purpose we define

\[
i_t = \frac{(-1)^{i-1}\tau_i}{\left[ \sum_{i=1}^{p} \tau_i^2 \right]^{1/2}} \tag{104}
\]

Then the vector

\[
t = (i_t, 2t \ldots pt) \tag{105}
\]

is the desired unit normal vector.
Transformation to v-space

To determine the distribution on the surface of $S_Y$, we introduce the following transformation from the $x$-space to the $v$-space:

$$x_i = i_f^i + v_1^i t \quad i = 1 \ldots p \quad (106)$$

Here the $i_f$ as given in (101) and the $i_t$ as defined by (104) are functions of $v_2 \ldots v_p$ and $v_1$ is a $p^{th}$ variable denoting distance along the normal vector $t$.

Lemma: The transformation defined by equations (106) is a one-to-one mapping of $\tilde{S}_Y$, the exterior of the convex region $S_Y$ and its boundary, to a subspace $\tilde{V}$ of the $v$-space. Further, the variables $v_2 \ldots v_p$ have the range of the surface parameters in describing the surface in (101) and for any point $(v_2 \ldots v_p)$, $v_1$ has the range of $0 \leq v_1 < \infty$.

Proof: Since the surface defined by (101) is convex, it follows that it is a simple surface, that is, the transformation defined by (101) is one-to-one. (We note that for $v_1 = 0$, equations (106) map the surface of $S_Y$ into a subspace of the hyperplane $v_1 = 0$.) It follows that, for a given point $v = (v_1 \ldots v_p)$ in $\tilde{V}$ for which equations (106) are defined, we get a unique point $x = (x_1 \ldots x_p)$ in $\tilde{S}_Y$. To show that the converse is true, i.e., that to a given point $x$ in $\tilde{S}_Y$ there corresponds a unique $v$ in $\tilde{V}$, assume the contrary. Thus, assume that two distinct points $v'$ and $v''$ in $\tilde{V}$
correspond to the same point \( x' = x'' \) in \( \hat{S}_Y \) under the mapping (106). Now for any point \( v \) in \( \hat{V} \), the coordinates \((v_2 \ldots v_p)\) determine a point on the surface of \( S_Y \) and also the normal vector at that point. The coordinate \( v_1 \) determines the distance along the normal from the surface to the point \( x \). This point is outside of \( S_Y \) for \( v_1 > 0 \) by virtue of the way we have defined the normal vector. Thus, if \((v_2' \ldots v_p') = (v_2'' \ldots v_p'')\) but \( v' \neq v'' \) then \( v' \) and \( v'' \) correspond to different points along the same normal line and hence \( x' \neq x'' \). If \((v_2' \ldots v_p') \neq (v_2'' \ldots v_p'')\) then these vectors determine two different points on the surface and, hence, two normal vectors. Thus, the equality of \( x' \) and \( x'' \) implies the existence of two distinct normals to the surface \( S_Y \) from an exterior point which is impossible by the convexity of \( S_Y \).

If we apply the transformation (106) to the distribution (100), hereafter omitting the \((\cdot)\), we have as the joint distribution of \( v_1 \ldots v_p \):

\[
g(v_1 \ldots v_p) = (2\pi\sigma^2)^{-p/2} \exp\left[-\frac{1}{2\sigma^2} \sum_{i=1}^{p} (i'f + v_1 i)^2\right] J \quad (107)
\]

Here \( J \) is the Jacobian of the transformation which is given symbolically by

\[
J = \frac{\partial(x_1 \ldots x_p)}{\partial(v_1 \ldots v_p)}.
\]
It will be convenient later to have the $v_1$ variable isolated and for this reason we rewrite the exponent in (107). Expanding, we have

$$
\sum_{i=1}^{p} (i^{\mathbf{f} + v_1 \mathbf{t}})^2 = \sum_{i=1}^{p} i^{\mathbf{f}^2 + 2v_1 \sum_{i=1}^{p} i^{\mathbf{f} \mathbf{t}} + v_1^2 \sum_{i=1}^{p} i^2} \tag{108}
$$

From equation (104) we see that $\sum_{i=1}^{p} i^2 = 1$ and hence by completing the square in the last two terms of (108) we have

$$
\sum_{i=1}^{p} (i^{\mathbf{f} + v_1 \mathbf{t}})^2 = \sum_{i=1}^{p} i^{\mathbf{f}^2} + (v_1 + \sum_{i=1}^{p} i^{\mathbf{f} \mathbf{t}})^2 - (\sum_{i=1}^{p} i^{\mathbf{f} \mathbf{t}})^2 \tag{109}
$$

For brevity we write

$$
R^2 = \sum_{i=1}^{p} i^{\mathbf{f}^2} - (\sum_{i=1}^{p} i^{\mathbf{f} \mathbf{t}})^2 \tag{110}
$$

$$
D = \sum_{i=1}^{p} i^{\mathbf{f} \mathbf{t}}
$$

Using (109) and (110) we may write (107) as

$$
g(v_1 \ldots v_p) = (2\pi \sigma^2)^{-p/2} \exp[-\frac{R^2}{2\sigma^2}] \exp[-\frac{(v_1 + D)^2}{2\sigma^2}]J \tag{111}
$$

The expression for the distribution on the surface of $\Sigma$ can now be obtained as a function of the surface coordinates $v_2 \ldots v_p$ by integrating $g(v_1 \ldots v_p)$ with respect to $v_1$ over the range of 0 to $\infty$. This integration is complicated by the
fact that the variable $v_l$ occurs in the Jacobian and hence it is necessary to determine precisely how the Jacobian may be written as a function of $v_l$.

**Expansion of the Jacobian**

By definition the Jacobian is

$$J(v_l) = \frac{\partial(x_1 \ldots x_p)}{\partial(v_1 \ldots v_p)}$$

(112)

Here we write $J(v_l)$ to emphasize that we are interested in expressing the Jacobian as a function of $v_l$ although we realize that it is a function of $v_1 \ldots v_p$. In (112) we are again using the notation

$$\frac{\partial i_f}{\partial v_j} = i_f^j \text{ and } \frac{\partial i_t}{\partial v_j} = i_t^j.$$  

For the particular case $v_1 = 0$ we have
If we expand the determinant in (113) and recall the definition of \( \tau \) and also the definition (104) of \( \tau \) we see that

\[
J(0) = \sum_{i=1}^{p} (-1)^{i-1} \tau_i^{\frac{1}{2}} = \left[ \sum_{i=1}^{p} \tau_i^{2} \right]^{1/2} = (114)
\]

It is clear that \( J(v_i) \) is a polynomial in \( v_i \) and a simple induction on \( p \) shows that \( J(v_i) \) is, in fact, a polynomial of degree \( p-1 \) in the variable \( v_i \). For the purpose of integrating \( g(v_i \ldots v_p) \) with respect to \( v_i \) it will be convenient to write this polynomial as

\[
J(v_i) = \sum_{i=0}^{p-1} b_i (v_i + D)^i = (115)
\]

where we have as yet to determine the coefficients \( b_i \). In order to determine the \( b_i \), we first express the polynomial \( J(v_i) \) in terms of the factorial powers of \( v_i \) [see, e.g., Richardson, 27, p. 10]. We have

\[
J(v_i) = J(0) + v_i (1) \frac{\Delta J(0)}{1!} + \ldots + v_i (i) \frac{\Delta^i J(0)}{i!} + \ldots + v_i (p-1) \frac{\Delta^{p-1} J(0)}{(p-1)!} (116)
\]
where \( v_1(i) = v_1(v_1 - 1) \ldots (v_1 - i + 1) \) and \( \Delta \) is the ordinary forward difference operator. Thus, \( \Delta J(0) = J(1) - J(0) \)
and in general \( \Delta^i J(0) = \sum_{h=0}^{i} (-1)^{i-h} \binom{i}{h} J(h) \). If we let \( c_i = \frac{\Delta^i J(0)}{i!} \)
then the polynomial (116) may be written more compactly as

\[
J(v_1) = \sum_{i=0}^{p-1} c_i v_1(i) \tag{117}
\]

where we adopt the convention that \( v_1(0) = 1 \) and \( \Delta^0 J(0) = J(0) \). The \( c_i \) may now be determined if we expand (112) for \( v_1 = 0, 1 \ldots p-1 \). The determination of the \( b_i \) in equation (115) will be complete if we now express the factorial powers of \( v_1 \) in terms of ordinary powers of \( v_1 + D \).

It is well known [see, e.g., Richardson, 27, p. 36] that the relation between factorial powers and ordinary powers is given in terms of Stirling Numbers of the First Kind. The relation being:

\[
v_1(i) = \sum_{k=1}^{i} S_k^i v_1^k \tag{118}
\]

We recall that these numbers may be generated from the recurrence relation \( S_k^{i+1} = S_{k-1}^i - i S_k^i \) with the conditions that \( S_0^i = 0, S_1^i = 1 \) and \( S_k^i = 0 \) for \( k > i \).

The expression for \( v_1(i) \) in powers of \( v_1 + D \) is now
easily obtained by expanding the right hand side of (118) about $v_1 = -D$. Using the fact that

$$v_1^k = \sum_{j=0}^{k} \binom{k}{j} (-D)^{k-j} (v_1 + D)^j \quad (119)$$

we get

$$v_1(i) = \sum_{k=1}^{i} \sum_{j=0}^{k} \binom{k}{j} (-D)^{k-j} S_k^i (v_1 + D)^j \quad (120)$$

We may now write the complete expression for $J(v_1^i)$ by combining (117) and (120). Recalling our definition of $c_i^e$, we have

$$J(v_1^i) = \sum_{i=0}^{p-1} \sum_{h=0}^{i} \sum_{k=1}^{i} (-1)^{i+k-h-j} \frac{1}{i!} \binom{i}{j} D^{k-j} J(h) S_k^i (v_1 + D)^j \quad (121)$$

The final expression for $J(v_1^i)$, although somewhat formidable in appearance, is relatively easy to obtain for a given surface point $(v_2 \ldots v_p)$ since the $S_k^i$ are easily generated. The major problem, of course, is the evaluation of the $J(h)$ for $h = 0, 1 \ldots p-1$.

A somewhat shorter expression is available in terms of Bernoulli polynomials by using the relation between factorial powers of $v_1$ and ordinary powers of $v_1 + D$ as given in Kendall [22, p. 58]. Thus in place of equation (120) we
may use

\[ v_1(i) = \sum_{j=0}^{i} \binom{i}{j} \frac{1}{i-j} (v_1 + D)^j (1 - D) (v_1 + D)^j \]  (122)

Here \( B_n(x) \) is called the Bernoulli polynomial of order \( n \) and degree \( r \) and is defined as the coefficient of \( t^r/r! \) in the series expansion of \( \left( \frac{t}{e^t - 1} \right)^n \). This expression (122), although somewhat simpler in appearance, requires that we have the Bernoulli polynomials available as there does not seem to be an easy way to generate them. For reference we list the first four here.

\[
\begin{align*}
B_0^n(x) &= 1 \\
B_1^n(x) &= x - \frac{n}{2} \\
B_2^n(x) &= x^2 - nx + \frac{n^2}{4} - \frac{n}{2} \\
B_3^n(x) &= x^3 - \frac{3n}{2} x^2 + 3\left(\frac{n^2}{4} - \frac{n}{12}\right)x + \frac{n^2}{8} (1 - n)
\end{align*}
\]

Integration with respect to \( v_1 \)

Returning to the problem of determining the surface distribution we recall that it is necessary to evaluate the integral
$$h(v_2 \ldots v_p) = \int_{0}^{\infty} g(v_1 \ldots v_p) \, dv_1 \quad (123)$$

For convenience we use expression (122) with equation (117) to give the following form for the Jacobian:

$$J(v_1) = \prod_{i=0}^{p-1} c_i \sum_{j=0}^{i} \binom{i}{j} B_{i-j}^1 (1 - D) (v_1 + D)^j \quad (124)$$

We recall that in expression (111) for the function $g(v_1 \ldots v_p)$ the quantity $R^2$ did not contain the variable $v_1$. To keep the following expressions as simple as possible we rewrite (111) in the form

$$g(v_1 \ldots v_p) = K \exp\left[-\frac{(v_1 + D)^2}{2\sigma^2}\right] J(v_1) \quad (125)$$

where $K = (2\pi\sigma^2)^{-p/2} \exp[-\frac{R^2}{2\sigma^2}]$. The integral (123) is then given by:

$$K \prod_{i=0}^{p-1} c_i \sum_{j=0}^{i} \binom{i}{j} B_{i-j}^1 (1 - D) \int_{0}^{\infty} \exp\left[-\frac{(v_1 + D)^2}{2\sigma^2}\right] (v_1 + D)^j \, dv_1 \quad (126)$$

If we focus our attention on the integral in the last expression and make the change of variable $z = \frac{v_1 + D}{\sigma}$ we see that this integral can be written as
\[
(2\pi)^{1/2} \int_0^\infty (2\pi)^{-1/2} \exp \left[-\frac{z^2}{2}\right] z^j \, dz \quad \text{D/\sigma}
\]
\[
= (2\pi)^{1/2} \sigma^{j+1} \mu_j(D/\sigma, \infty)
\]

Here \(\mu_j(D/\sigma, \infty)\) denotes the incomplete normal moment function which is tabulated in Pearson [26, pp. 22-23].

In summary we see that the surface distribution is given as a function of \(v_2 \ldots v_p\) as

\[
h(v_2 \ldots v_p) = K \sum_{i=0}^{p-1} c_i \sum_{j=0}^{i} (iB_{i-j}^j (1-D)(2\pi)^{1/2} \sigma^{j+1} \mu_j(D/\sigma, \infty)
\]

The complexity of this exact distribution limits its use since the determination of the moments is obviously not a simple problem in general. In the next section we illustrate this with a simple example.

**Example**

As an illustration of this distribution we consider the situation in which the restraining region is a sphere of radius \(r\) about the origin of the \(\gamma\)-space. The parametric equations corresponding to (101) which describe the sphere for the case \(p = 2\) are
\[
x_1 = f(v_2) = r \cos v_2 - 0Y_1 \\
x_2 = f(v_2) = r \sin v_2 - 0Y_2
\]  
(129)

Here we have made the transformation (99) so that the true parameter point is at the origin of the x-space. In the notation used in the general development we see that the matrix \(M\) of equation (102) is

\[
M = \begin{bmatrix}
- r \sin v_2 \\
 r \cos v_2
\end{bmatrix}
\]  
(130)

By definition of \(i^t\), equation (104), we see that \(i^t = \cos v_2\) and \(2^t = \sin v_2\) and hence the transformation (106) to the \(v\)-space is given by

\[
x_1 = r \cos v_2 + v_1 \cos v_2 - 0Y_1 \\
x_2 = r \sin v_2 + v_1 \sin v_2 - 0Y_2
\]  
(131)

The Jacobian of the transformation is

\[
J(v_1) = \begin{vmatrix}
\cos v_2 & \sin v_2 \\
-(r + v_1) \sin v_2 & (r + v_1) \cos v_2
\end{vmatrix} = r + v_1
\]  
(132)
From equations (110) we see that after some simplifications, $R^2 = (0y_1 \sin v_2 - 0y_2 \cos v_2)^2$ and $D = r - 0y_1 \cos v_2 - 0y_2 \sin v_2$. Using these quantities, the expression corresponding to equation (128) for the surface distribution is given by

\[
h(v_2) = K \int_0^\infty (v_1 + r) \exp \left[ \frac{1}{2\sigma^2} (v_1 + r - 0y_1 \cos v_2 - 0y_2 \sin v_2)^2 \right] dv_1
\]

\[
= K [\sigma^2 \exp \left[ \frac{1}{2\sigma^2} (r - 0y_1 \cos v_2 - 0y_2 \sin v_2)^2 \right] ]
\]

\[
+(0y_1 \cos v_2 + 0y_2 \sin v_2) \int_0^\infty \exp \left[ \frac{1}{2\sigma^2} (v_1 + r - 0y_1 \cos v_2 - 0y_2 \sin v_2)^2 \right] dv_1
\]

(133)

Clearly the determination of the moments of $\gamma$ in this case is a difficult problem. For the purpose of illustration we consider the special case in which the true parameter point is at the origin of the sphere. In this case $R^2 = 0$ and $D = r$ hence the distribution on the surface corresponding to equation (133) is given by

\[
h(v_2) = (2\pi \sigma^2)^{-1} \int_0^\infty \exp \left[ - \frac{(v_1 + r)^2}{2\sigma^2} \right] (v_1 + r) dv_1
\]

\[
= \frac{1}{2\pi} \exp \left[ - \frac{r^2}{2\sigma^2} \right]
\]

(134)
To determine the moments of $\tilde{x}$ it is necessary to perform two integrations as indicated in the following:

$$E[x_1^k x_2^h] = \int \int x_1^k x_2^h N(x : 0, \sigma^2 I) \, dx_1 \, dx_2$$

$$F(x) \leq 0$$

$$+ \int x_1^k x_2^h h(v_2) \, dv_2$$

$$F(x) = 0$$

The first integration in (135) is taken over the interior or the constraining sphere with $N(x : 0, \sigma^2 I)$ indicating the ordinary spherical normal distribution centered at the origin. The second integration is to be taken over the surface of the sphere.

In this case it is intuitively obvious that $\tilde{y}$ is unbiased since $\gamma$ is unbiased and the projection to the sphere is along radial lines emanating from the center which is $\gamma_0$. This intuition is easily checked by integration of (135) which shows that $E(x_1) = E(x_2) = 0$. We now proceed to calculate the second moments. From equation (135) it follows that

$$E(x_1^2) = 4 \int \int_0^r (r^2 - x_2^2)^{1/2} x_1 x_2 \frac{1}{2\pi \sigma^2} \exp[- \frac{(x_1^2 + x_2^2)}{2\sigma^2}] \, dx_1 \, dx_2$$

$$+ \int_0^{2\pi} r^2 \cos^2 v_2 \frac{1}{2\pi} e^{-r^2/2\sigma^2} \, dv_2$$

$$= \sigma^2 (1 - e^{-r^2/2\sigma^2})$$
Similarly we see that \( E(x_2^2) = E(x_1^2) \) and \( E(x_1 x_2) = 0 \). Thus the covariance matrix of \( \hat{\gamma} \) is seen to be \( \sigma^2(1 - e^{-r^2/2\sigma^2})I_2 \) and we notice a reduction in variance as compared with the \( \hat{\gamma} \) estimator.

This contrived example obviously oversimplifies the general situation but does suggest that the distribution may be of some value. Some further investigations into the classes of regions which lead to relatively simple expressions for the moments may provide us with sufficient information to give us good approximations for the moments in a given problem. Thus, depending on the nature of the region \( S_\gamma \) and on the inference which can be made on \( \gamma_0 \) we may be able to approximate moments which will be better than the previously developed asymptotic moments.

**Extensions**

The distribution theory discussed in the previous section was restricted to the situation where the constraining region was bounded by a single smooth surface. The more general situation, in which the region \( S_\gamma \) is the intersection of several convex regions, becomes more complicated. The main reason for this complication is the fact that along the intersections we have a higher dimensional loading of probability. That is to say, whenever we are at a point in the intersection of two or more surfaces, we must sum over a
region in space rather than simply summing along a line normal to the surface.

If we restrict the discussion to the case \( p = 2 \), we have a relatively simple problem and it will serve to indicate what is involved in the case of more than two regression coefficients. For the present let us assume that the convex region \( S \) is determined as the intersection of just two convex regions defined by \( F(\gamma) \leq 0 \) and \( G(\gamma) \leq 0 \). Let us suppose that the portion of the boundary of \( S \) defined by \( F(\gamma) = 0 \) is given parametrically by \( \gamma_i = \int f(\theta), i = 1, 2 \), when the parameter \( \theta \) ranges over some one-dimensional region. Similarly, let the portion of the boundary defined by \( G(\gamma) = 0 \) be given parametrically by \( \gamma_i = \int g(\lambda), i = 1, 2 \). Let us refer to the points of intersection of the two curves, \( F(\gamma) = 0 \) and \( G(\gamma) = 0 \), as vertices of the region \( S \). If, for the moment, we exclude these vertices, the probability distribution on the surface of \( S \) can be obtained by two transformations of the type (106), one for each of the surfaces.

Let us denote the vertices of \( S \) by \( \gamma' \) and \( \gamma'' \). To determine the probability mass at a particular vertex it is necessary to determine the probability content of the cone defined by the normal vectors to the two curves at the given vertex. Thus, for example, at the vertex \( \gamma' \) we have the two normal vectors given by \( \left( \frac{\partial F}{\partial \gamma_1}, \frac{\partial F}{\partial \gamma_2} \right) \) and \( \left( \frac{\partial G}{\partial \gamma_1}, \frac{\partial G}{\partial \gamma_2} \right) \) where the
the partial derivatives are to be evaluated at $\gamma'$. The integration of the spherical normal over this cone will give the probability mass at the vertex $\gamma'$.

In general, the integration of the spherical normal over a region of this type does not result in a closed expression in terms of well known functions. Ruben [29] devotes some attention to this problem.

The extension of this notion to more than two defining curves is obvious but the case $p \geq 3$ becomes more difficult and we shall not pursue it here.
Preliminary remarks

We have seen that if the constraining region \( S_\beta \) is defined by differentiable convex functions, the determination of \( \hat{\beta} \) can be achieved by the algorithm developed in Part I. In this section we propose to develop a quadratic programming procedure which will solve the slightly more general problem in which we require only that the region \( S_\beta \) be convex. Thus, we allow the possibility of having the boundary defined by quasi-convex functions.

Let us assume that we have already transformed to the canonical parameter space so that the linear model with which we are working is \( y = \mathbf{U} \gamma + e \) as defined in equation (64). Then by (71) we see that the solution to the constrained regression problem is obtained by finding that point in \( S_\gamma \) which is "nearest" the least squares point \( \hat{\gamma} \). For certain types of restraining regions this problem is quite simple. For example, if \( S_\gamma \) is defined by a single quadratic function then the problem is easily reduced to solving one equation in a single variable.

In general, the solution is not so easily obtained and we suggest here an iterative procedure for determining \( \gamma \). We discuss in detail an algorithm for the case in which \( S_\gamma \) is bounded by a single surface and suggest later how this
procedure can be extended to more complex regions.

To formulate the problem more definitely, we assume that the region $S_Y$ is defined to be the set of all points $\gamma$ for which $F(\gamma) \leq 0$ where $F(\gamma)$ is a quasi-convex, differentiable function. We assume that the least squares point $\hat{\gamma}$ has already been determined and that $\hat{\gamma}$ is outside of $S_Y$, i.e., $F(\hat{\gamma}) > 0$. Of course, if $F(\hat{\gamma}) \leq 0$ then by (71) we have $\hat{\gamma} = \gamma$ and the problem is solved. We assume also that a point $Q_Y$ is available for which $F(Q_Y) < 0$. That is, a feasible point $O_Y$ is known from a priori considerations.

The algorithm

In principle this algorithm is quite simple and we outline it briefly first so as to make it easier to follow the detailed discussion. The steps of the algorithm are:

(i) Determine the point $\tilde{\gamma}$ which is at the intersection of the surface $F(\gamma) = 0$ with the line from the feasible point $O_Y$ to the point $\hat{\gamma}$.

(ii) Determine the tangent plane to $F(\gamma) = 0$ at the point $\tilde{\gamma}$ and then determine the point $\gamma^*$ in this plane which is at the foot of the perpendicular from $\hat{\gamma}$ to the plane.

(iii) If $\gamma^* = \tilde{\gamma}$, then $\gamma = \gamma$. If $\gamma^* \neq \tilde{\gamma}$, search the surface of $S_Y$ in the plane of $\gamma$, $\gamma^*$, $\gamma$ for the point nearest $\hat{\gamma}$. Call this point $2\gamma$.

(iv) Repeat steps (ii) and (iii) with $\gamma$ replaced by $2\gamma$ and
in general with \( j \gamma \) replaced by \( j+1 \gamma \).

We look now at the details involved in these steps so that we may state the algorithm more precisely. In parametric form the line from \( 0 \gamma \) to \( \hat{\gamma} \) is given by

\[
\gamma_i(\lambda) = \gamma_j + \lambda(\hat{\gamma} - \gamma_j)
\]

To determine the point \( \gamma \), it is necessary to determine the intersection of this line with the surface \( F(\gamma) = 0 \). Substituting into this equation the expression for \( \gamma_i(\lambda) \) from equation (137) we get the equation \( F(\gamma(\lambda)) = 0 \) which is a function of the single parameter \( \lambda \). We determine the unique solution \( \lambda' \) in the range \( 0 < \lambda < 1 \) by some method of inverse interpolation such as the Newton-Raphson iteration [see, e.g., Hildebrand, 17, p. 443]. The coordinates of the first trial solution \( \gamma \) are then given by equation (137) as \( \gamma = \gamma_i(\lambda') \).

If we denote \( \frac{\partial F(\gamma)}{\partial \gamma_i} \) evaluated at the point \( \gamma \) by \( F_i(\gamma) \), then a set of direction numbers for the normal line at \( \gamma \) are given by

\[
F_1(\gamma) : F_2(\gamma) : \ldots : F_p(\gamma)
\]
\[ \sum_{i=1}^{p} F_i(y) (\gamma_i - y_i) = 0 \]  

(139)

To express equation (139) in a more abbreviated form we let
\[ F_i(y) = a_i \]  
and
\[ \sum_{i=1}^{p} F_i(y) \gamma_i = a' \gamma = h. \]  
The equation of the tangent plane can then be written in vector notation as
\[ a' \gamma = h \]  

(140)

We note that from the way in which \( \gamma \) was determined it follows that \( a' \gamma \geq h. \)

Several methods may be used to determine the point \( \gamma^* \) which is at the foot of the perpendicular from \( \gamma \) to the plane given by equation (140). As a matter of interest and for the purpose of later generalization we state this problem as one in quadratic programming. In particular, we look for that vector \( \gamma^* \) which minimizes the quadratic form
\[ (\gamma - \hat{\gamma})' (\gamma - \hat{\gamma}) \]  

(141)

subject to the linear restriction
\[ a' \gamma \leq h \]  

(142)

Clearly, the solution to this problem is the required foot point \( \gamma^* \).
Now the Kuhn-Tucker Theorem which we reviewed briefly in Part I says that this problem is equivalent to the saddle-point problem

$$\min_{\rho} \max_{\gamma} L(\rho, \gamma) \quad \rho \geq 0 \quad (143)$$

where $L(\rho, \gamma)$ is the Lagrangian function which in this case is

$$L(\rho, \gamma) = -(\gamma - \hat{\gamma})'(\gamma - \hat{\gamma}) + \rho(h - a'\gamma) \quad (144)$$

Differentiating (144) with respect to $\gamma$ we have

$$\frac{\partial L}{\partial \gamma} = -2(\gamma - \hat{\gamma})' - \rho a'$$

and hence for a given $\rho$ we see that the maximum value of $L(\rho, \gamma)$ is given by

$$\gamma^m = \hat{\gamma} - \frac{1}{2} \rho a \quad (145)$$

Substituting this expression into equation (144) we have

$$L(\rho, \gamma^m) = -\frac{1}{4} \rho^2 a'a + \rho[h - a'(\hat{\gamma} - \frac{1}{2} \rho a)]$$

$$= \frac{1}{4} \rho^2 a'a + \rho(h - a'\hat{\gamma}) \quad (146)$$

If we call this function $\phi(\rho)$, then we see that the quadratic programming problem given by equations (141) and (142) has been reduced to finding the value of $\rho$ which will minimize
\& \rho(p) \text{ subject to the condition that } \rho \geq 0. \text{ The solution is thus given by}
\[
\rho = \begin{cases} 
2 \left( \frac{h - a' \cdot \hat{\gamma}}{a' a} \right) & \text{if } \rho > 0 \\
0 & \text{if } \rho \leq 0
\end{cases} 
\tag{147}
\]

From equation (145) we see that the solution to the problem (141) and (142) is given by
\[
\gamma^* = \begin{cases} 
\hat{\gamma} + \left( \frac{h - a' \cdot \gamma}{a' a} \right) a & \text{if } \rho > 0 \\
\gamma & \text{if } \rho \leq 0
\end{cases} 
\tag{148}
\]

To assure ourselves that this is the correct solution we note that \(a/(a' a)^{1/2}\) is the unit vector normal to the surface at \(\gamma^*\). Then, since \(h - a' \cdot \hat{\gamma} = -a' (\gamma - \gamma^*)\) it follows that \((a' \cdot \gamma - h)/a' a\) is the projection of the vector from \(\gamma^*\) to \(\gamma\) onto the unit normal vector \(a/(a' a)^{1/2}\). This must be subtracted from \(\gamma\) to yield \(\gamma^*\) which is the result of equation (148). The case \(\gamma^* = \gamma\) occurs when the tangent plane at \(\gamma\) passes through \(\gamma^*\).

We now proceed with the more difficult part of the algorithm, that is, the determination of the next trial solution \(\gamma^*\). For the moment we assume \(\rho > 0\) in (148), and treat
the case \( \hat{\gamma} = 1\gamma^* \) later as a special case. As previously mentioned, the next trial solution \( 2\gamma \) is that point on the surface of \( S_\gamma \) in the plane of \( 1\gamma, 1\gamma^*, \hat{\gamma} \) which is closest to \( \hat{\gamma} \). Clearly, if \( 1\gamma^* \neq 1\gamma \), there is such a point \( 2\gamma \) which satisfies these restrictions and is closer to \( \hat{\gamma} \) than is the point \( 1\gamma \). Reference to Figure 1 may help guide the intuition.

Figure 1. Illustration of the algorithm in two-dimensions
The equation of the line from \( y \) to \( y^* \) in parametric form is

\[
\gamma_1(\lambda) = y_1 + \lambda(\gamma_1^* - y_1)
\]

\[
= (1 - \lambda) y_1 + \lambda y_1^*
\]

(149)

and by the convexity of the region \( S_\gamma \) we know that a line from \( \gamma \) to the new trial point \( 2y \) will intersect the line (149) for some \( 0 < \lambda \leq 1 \). For convenience let us write

\[
e = (h - a') \gamma (a'a)^{-1/2}
\]

(150)

Then the vector \( y^* = \gamma + e(a'a)^{-1/2}a \) and we may rewrite equation (149) as

\[
\gamma_1(\lambda) = (1 - \lambda) y_1 + \lambda(\gamma_1 + e(a'a)^{-1/2}a_i)
\]

(151)

Let \( \gamma(\lambda') \) denote a point on this line for some \( 0 \leq \lambda' \leq 1 \). Then the equation of the line from \( \gamma \) to \( \gamma(\lambda') \) may be written parametrically as

\[
\gamma_i(\eta) = \gamma_i + \eta(\gamma_i(\lambda') - \gamma_i)
\]

(152)

In general then, the equation of the line from \( \gamma \) to any point on the line joining \( y \) to \( y^* \) may be written as a function of
the two parameters $\lambda$ and $\eta$ by combining (151) and (152) to get

$$\lambda_i(\eta, \lambda) = \gamma_i + \eta[(1 - \lambda)(\gamma_i - \gamma_1) + \lambda e(a' a)^{-1/2}a_i] \quad (153)$$

We note that with $\lambda = 0$ this line (153) is just the line from $\gamma$ to $\gamma$ and hence for some $\lambda > 0$ this line (153) will intersect the surface. The points of intersection of the line (153) with the surface $F(\gamma) = 0$ are determined by those pairs of parameter values $(\eta, \lambda)$, $0 \leq \lambda \leq 1$, $\eta \geq 1$, which satisfy the equation

$$F[\gamma(\eta, \lambda)] = H(\eta, \lambda) = 0 \quad (154)$$

We pause a moment to consider the special case $\gamma = \gamma^*$ which we have excluded thus far. In this case we do not have the necessary three points to determine the plane in which we search for $\gamma$ and the line (153) degenerates into a one-parameter line from $\gamma$ to $\gamma$. To overcome this difficulty we designate a pseudo point $\gamma^{**}$ to play the role of $\gamma^*$ as follows:

$$\gamma^{**} = \gamma - a \quad (155)$$

Proceeding as above we see that in this case the equation
analagous to (153) is

$$\gamma_i(\eta, \lambda) = \hat{\gamma}_i + \eta[(1 - \lambda)(\hat{\gamma}_i - \hat{\gamma}_i) - \lambda a_i] \quad (156)$$

The discussion given for the general case holds here except for the fact that the value of \( \eta \) corresponding to a point of intersection with \( F(\gamma) = 0 \) may be less than one. Clearly, this case, \( \hat{\gamma} = \gamma^* \), has low probability of occurrence.

Returning to the general discussion, we see that the problem of determining \( \hat{\gamma} \) is now phrased as the problem of determining that pair of parameters \((\eta', \lambda')\) which satisfy equation (154) and minimize the distance from \( \hat{\gamma} \) to \( \gamma(\eta', \lambda') \). More precisely we wish to find the pair \((\eta, \lambda)\) which minimize

$$\sum_{i=1}^{p} [\gamma_i(\eta, \lambda) - \hat{\gamma}_i]^2 \quad (157)$$

subject to the condition that

$$H(\eta, \lambda) = 0 \quad (158)$$

Here \( H(\eta, \lambda) \) is defined by equation (154), and \( \gamma_i(\eta, \lambda) \) is given by equation (153). If we apply the method of Lagrange multipliers to the problem, the Lagrangian function is given by
Before proceeding, we expand \( \sum_{i=1}^{p} [\gamma_i(\eta, \lambda) - \hat{\gamma}_i]^2 \) in terms of \( \eta \) and \( \lambda \) and then simplify as follows:

\[
\sum_{i=1}^{p} [\gamma_i(\eta, \lambda) - \hat{\gamma}_i]^2 = \sum_{i=1}^{p} \eta^2[(1-\lambda)(\gamma_i - \hat{\gamma}_i) + \lambda e(a'a)^{-1/2}a_i]^2
\]

\[
= \eta^2[(1-\lambda)^2 \sum_{i=1}^{p} (\gamma_i - \hat{\gamma}_i)^2 + \lambda^2e^2
\]

\[
+ 2\lambda e(a'a)^{-1/2}(1-\lambda) \sum_{i=1}^{p} a_i(\gamma_i - \hat{\gamma}_i)]
\]

\[
= \eta^2[(1-\lambda)^2 d^2 + \lambda^2e^2 + 2\lambda(1-\lambda)e^2]
\]

\[
= \eta^2[(1-\lambda)^2 (d^2 - e^2) + e^2]
\]

(160)

Here we have let \( d^2 = \sum_{i=1}^{p} (\gamma_i - \hat{\gamma}_i)^2 \), the squared distance from \( \gamma \) to \( \hat{\gamma} \) and we have noted that

\[
(a'a)^{-1/2} \sum_{i=1}^{p} a_i(\gamma_i - \hat{\gamma}_i) = (a'a)^{-1/2}a'(\gamma - \hat{\gamma}) = e
\]

by equation (150). We note also that \( e^2 \) is the squared distance from \( \hat{\gamma} \) to \( \gamma^* \).

Using expression (160) in the Lagrange function (159)
and differentiating with respect to each of the parameters \( \eta, \lambda \) and \( \alpha \) we have

\[
\frac{\partial R}{\partial \eta} = 2\eta[(1 - \lambda)^2 (d^2 - e^2) + \alpha \frac{\partial H}{\partial \eta}]
\]

\[
\frac{\partial R}{\partial \lambda} = 2\eta^2 (\lambda - 1) (d^2 - e^2) + \alpha \frac{\partial H}{\partial \lambda}
\]

(161)

\[
\frac{\partial R}{\partial \alpha} = H(\eta, \lambda)
\]

The solution is obtained by setting the three expressions in (161) equal to zero and solving simultaneously. Obviously, we can reduce the number of equations to two by eliminating \( \alpha \) between the first two. The result of this elimination is

\[
T(\eta; \lambda) = \eta(d^2 - e^2)(\lambda - 1) \frac{\partial H}{\partial \eta} - [(d^2 - e^2)(\lambda - 1)^2 + e^2] \frac{\partial H}{\partial \lambda} = 0
\]

(162)

Thus we need only solve the two nonlinear equations (158) and (162), i.e., \( H(\eta, \lambda) = 0 \) and \( T(\eta; \lambda) = 0 \).

As we have seen, the solution will be such that \( 0 \leq \lambda \leq 1 \) and \( \eta \geq 1 \). However, in general there may be more than one value of \( \eta \) for a given value of \( \lambda \). We are of course interested in the smallest value of \( \eta \). A bound can be placed on the correct value of \( \eta \) since it is clear that for the true
values of \( \eta \) and \( \lambda \) we have the inequality
\[ |\gamma(\eta, \lambda) - \hat{\gamma}|^2 \leq d^2. \]
Using this with expression (160) and the fact that \( 0 \leq \lambda \leq 1 \) we see that
\[
\eta^2 \leq \frac{d^2}{(1 - \lambda)^2 (d^2 - e^2) + e^2} \leq \frac{d^2}{e^2}
\]
and hence we have the bounds
\[
1 \leq \eta \leq \left| \frac{d}{e} \right| \quad (163)
\]
The solution of the nonlinear equations (158) and (162) may be carried out by any of the standard iterative procedures [see, e.g., Hildebrand, 17, p. 450] aided by the a priori knowledge of the bounds on \( \eta \) and \( \lambda \).
To summarize the iterative part of the algorithm, we see that to go from a trial solution \( k\gamma \) to the next solution \( k+1\gamma \) we proceed as follows:
(i) Determine the vector \( k a \) of partial derivatives of \( F(\gamma) \) at the point \( k\gamma \).
(ii) From the vectors \( \hat{\gamma}, k\gamma \) and \( k a \), determine
\[
d^2 = |k\gamma - \hat{\gamma}|^2
\]
and
\[
e = \frac{k a'(k\gamma - \hat{\gamma}) (k a'_k a)^{-1/2}}{k a}. \]
(iii) Determine the equations (158) and (162) for these particular quantities and solve for \( \eta \) and \( \lambda \).
(iv) Determine \( k+1\gamma \) from equation (153).
A numerical example

To illustrate this quadratic programming procedure we consider a simple numerical example. Suppose that we are already in the canonical space and that the least squares point \( \hat{\gamma} \) has been found to be, \( \hat{\gamma}_1 = 5, \hat{\gamma}_2 = 0 \). Let the convex region \( S_\gamma \) be defined by the single inequality

\[
F(\gamma) = \gamma_1^2 - \gamma_2 \leq 0
\]  
(164)

We note that, except for the way in which the problem has been stated, this is precisely the same problem as given in Part I, Example 2.

Let us assume that the feasible point \( \gamma = 3, 9 \) is available. We note that in this case we may let \( \gamma = \gamma \).

The vector of partial derivatives of \( F(\gamma) \) evaluated at \( \gamma \) is given by \( a' = (6, -1) \). From equation (150) we have

\[
e = \frac{a'(\gamma - \hat{\gamma})}{(a'a)^{1/2}} = \frac{-21}{\sqrt{37}}
\]

and we also have

\[
d^2 = |\gamma - \hat{\gamma}|^2 = 85
\]

In this case \( H(\eta, \lambda) \) is obtained by substituting the expression for \( \gamma(\eta, \lambda) \) from equation (153) into equation (164).
With this expression we are now able to write equation (162) for this particular problem. With the above values for e, d^2 and the vector a, we write these equations, after some simplification, as:

\[
\eta = 0.862069 - 0.193848 \eta \lambda \\
+ \eta^2[0.137931 + 0.193848 \lambda + 0.068109 \lambda^2]
\]

and

\[
\lambda = 1.026482 - 0.419924 \eta + 0.098361 \eta \lambda \\
+ 0.276473 \eta \lambda^2
\]

Here the first equation is obtained from \(H(\eta, \lambda) = 0\) and the second from \(T(\eta, \lambda) = 0\). Although there are more refined methods, we solved these equations by the method of successive substitutions starting with the initial point \(\lambda = 0, \eta = 1\). After thirty iterations we have \(\eta = 1.148383, \lambda = 0.909860\). From equation (153) we then determine the vector \(2\gamma\) which in the case \(p = 2\) is, in fact, \(\tilde{\gamma}\). The result being \(\tilde{\gamma}_1 = 1.23477, \tilde{\gamma}_2 = 1.52466\). Reference to Example 2, Part I shows that this answer is correct to five decimal places.
Comparison with the algorithm of Part I

The present algorithm has obvious computational disadvantages when compared with the general convex programming algorithm proposed in Part I. The necessity of transforming to the canonical space, the determination of $\gamma_0$, and the necessity of solving the pair of nonlinear equations at each step are obviously bad features of the procedure. In addition to these, this method has the troublesome feature of requiring an initial feasible point $\gamma_0$ in order to start the solution.

This quadratic algorithm is applicable to a slightly larger class of restraining functions however, and so should not be disregarded completely. Also in favor of the method is the fact that, at least intuitively, a rather small number of iterations will give a reasonably good solution. In fact, the first trial solution, $\gamma_1$, has certain good properties, e.g., it is consistent. As yet only simple examples have been tried and so no comparisons can be made as to the relative amounts of computation involved in the two algorithms in the case where both are applicable. That is, the case in which the boundary of $S_\gamma$ is defined by a single convex function.
Extension to more than one restraint function

Thus far we have considered only the situation in which the region \( S_\gamma \) is defined by a single quasi-convex function. In general, \( S_\gamma \) will be the set of points, \( \gamma \), which satisfy the \( m \) inequalities \( jF(\gamma) \leq 0, j = 1 \ldots m \), where the \( jF(\gamma) \) are quasi-convex functions. The algorithm developed above then fails since if we apply it at the point \( k\gamma \), the next trial solution, \( k+1\gamma \), may not satisfy \( jF(\gamma) \leq 0 \) for some \( j \). We may, of course, "correct back" to the feasible region but then we are at a point on the surface, call it \( k+1\gamma^c \), at which there may not be a unique tangent plane. That is, we are at a point of intersection of two or more surfaces. Paralleling the above algorithm, what we would like to have is a point, which we may call \( \gamma^* \) for analogy with the single surface procedure, such that if we search along the surface of \( S_\gamma \) in the plane of \( k+1\gamma^c \), \( \gamma^* \), \( \hat{\gamma} \) we are guaranteed to find a point \( k+2\gamma \) which is closer to \( \hat{\gamma} \) than is \( k+1\gamma^c \).

For definiteness let us assume that the point \( k+1\gamma^c \) lies on the intersection of \( q \) surfaces. Then we may consider the linear region bounded by the \( q \) tangent planes at \( k+1\gamma^c \). By direct extension of the problem given by (141) and (142) we determine the point \( \gamma^* \) by minimizing the quadratic form

\[
(\gamma - \hat{\gamma})'(\gamma - \hat{\gamma})
\]

(165)
subject to the linear restrictions

\[ H\gamma \leq h \quad \text{(166)} \]

Here the system of inequalities (166) is determined by the \( q \) tangent planes at \( k+l \gamma^C \). Without going into detail, we assert that if we let the solution of the above quadratic program play the role of \( \gamma^* \) in the single surface procedure, we will be led to an improved trial solution. Of course we have the additional problem of ensuring that the new trial solution remains in the feasible region.

The solution of the quadratic program given by (165) and (166) may be obtained by any of the standard methods. However, since the most frequently occurring situation is that in which there are only two surfaces intersecting at \( k+l \gamma^C \), we recommend the method of Hildreth [18]. The reason for this recommendation is that Hildreth reduces the problem (165) and (166) to the problem of determining the \( q \times 1 \) vector \( \rho^* \) which minimizes

\[ \varnothing(\rho) = \rho' B \rho + b' \rho \quad \text{(167)} \]

subject to the restriction \( \rho \geq 0 \). Here \( B = 1/4 \, HH' \) and \( b^- = h - H\gamma \). The solution to (165) and (166) is then given by
\[ y^* = \gamma - \frac{1}{2} H'p^* \]  \hspace{1cm} (168)

In the case \( q = 2 \), the quadratic program (167) may be solved in one step of the Hildreth procedure.

It is clear that this generalized quadratic programming algorithm is computationally quite inferior to the algorithm of Part I when the \( jF(\gamma) \) are convex functions. In addition to the disadvantages of the single surface procedure we now have a quadratic program to solve at each iteration and we must always ensure that the trial solutions are in the feasible region. However, in spite of the obvious computational difficulties, the method does have the advantages mentioned above for the single surface situation and so should be given some consideration.

The discussion given here for the case of more than one restraining surface is quite brief especially with regard to the way in which we search for the next trial solution. The procedure is quite straightforward however, and rather than spell out the details we give in the next section an example which should serve to illustrate the method.

**Example with two restraining surfaces**

We consider the problem of finding the point in the region defined by
\[ F(\gamma) = (\gamma_1 - 1)^2 + \gamma_2^2 + \gamma_3^3 \leq 1 \]
\[ 2F(\gamma) = 0.5 \gamma_1 - \gamma_2 \leq 0.125 \]

which is nearest to the point \( \hat{\gamma} \), where \( \hat{\gamma}_1 = 1 \), \( \hat{\gamma}_2 = 0 \), \( \hat{\gamma}_3 = 2 \). The algorithm proceeds as in the single surface situation until we arrive at a point on the intersection of two surfaces. (Note that the single surface procedure must be modified since we are not certain that the next trial solution is in the feasible region. In the following we suggest an alternate procedure for determining the next trial solution.) Since the procedure at an intersection point is of most interest, we choose an initial feasible point \( \gamma^0 \) at such a point, the coordinates being \( \gamma^1 = 0.25 \), \( \gamma^2 = 0 \), \( \gamma^3 = \sqrt{7}/4 = 0.661438 \). To determine the point \( \gamma^* \) we must solve the quadratic program given by equations (165) and (166). At the point \( \gamma^1 \) the matrix \( H \) and the vector \( h \) are given by

\[
H = \begin{bmatrix} -1.5 & 0 & 1.322874 \\ 0.5 & -1 & 0 \end{bmatrix}
\]

\[
h' = (0.5, 0.125)
\]

Using the method of Hildreth [18], we find the vector \( p \) which
minimizes

\[ \varnothing(p) = p' (1/4 \mathbf{H} \mathbf{H}') p + (h - \mathbf{H} \hat{\gamma})' p \]

subject to the condition \( p \geq 0 \). Throughout this solution \( p \) is given by \(-1/2 \mathbf{B}^{-1} \mathbf{b}\) where \( \mathbf{B} = 1/4 \mathbf{H} \mathbf{H}' \) and \( \mathbf{b} = h - \mathbf{H} \hat{\gamma} \), but this is not generally the case and the Hildreth procedure must actually be applied. At the point \( \hat{\gamma} \) the solution is \( p' = (0.490562, 0.894337) \) and hence \( \gamma^* = \hat{\gamma} - 1/2 \mathbf{H}' p^* \) is given by the row vector \( \gamma^* = (1.144338, 0.447169, 1.675524) \). It is clear that if we search each of the surfaces which intersect at \( \hat{\gamma} \) in the plane of \( \hat{\gamma}, \gamma, \gamma^* \) then at least one of them will yield a feasible point \( 2\gamma \) which is closer to \( \hat{\gamma} \) than is \( \gamma^* \). In this case we consider the sphere defined by \( 1F(\gamma) = 1 \). To determine the next point \( 2\gamma \), we could use the single surface procedure and solve the equations (158) and (162), however, we are not assured that this point is feasible and so we use the following technique:

The parametric equations of the line from \( \hat{\gamma} \) to a point on the sphere in the plane of \( \hat{\gamma}, \gamma, \gamma^* \) are given by combining equations (149) and (152) to get

\[ \gamma_i(\eta, \lambda) = \hat{\gamma}_i + \eta(\gamma_i - \hat{\gamma}_i) + \lambda(\gamma^*_i - \gamma_i) \]

Substituting these into the equation \( 1F(\gamma) = 0 \) we get an
equation in \( \eta \) and \( \lambda \). By a double-interpolation technique we determine pairs \((\eta, \lambda)\) which satisfy this equation and choose that pair which minimizes \(|\gamma_1^*(\eta, \lambda) - \hat{\gamma}|^2\) subject to the condition that \(\gamma_1^*(\eta, \lambda)\) is in the feasible region.

In this case the result is \(\eta = 1.47096, \lambda = .550\) and hence \(\gamma\) is given by the vector \(\gamma' = (.620324, .361773, .851451)\). The point \(\gamma\) is only in one of the surfaces and hence the single surface procedure may be applied. Again we must ensure feasibility, and so the double-interpolation technique is preferable to solving the equations (158) and (162).

Continuing in this manner we generate a sequence of trial vectors, \(1\gamma, 2\gamma, \ldots\), each closer to \(\gamma\) than the previous one, and hence we converge on the true point \(\gamma\).

We note that the essential feature of this method is that it gives us a scheme for moving off of the intersections so that we can apply the single surface technique and that it does so in such a way that the next trial solution is closer to \(\gamma\) than is the present one.

In Table 4, we list the trial solutions \(1\gamma\) for the first seven iterations and we also give the squared distance from \(\gamma\) to \(1\gamma\). The solution correct to six places in \(|1\gamma - \hat{\gamma}|^2\) is \(1\gamma = .850, 2\gamma = .300, 3\gamma = .942072, |1\gamma - \hat{\gamma}|^2 = 1.231711\).
Table 4. Trial solutions for the quadratic programming problem

|  \( \hat{\gamma}_1 \) |  \( \hat{\gamma}_2 \) |  \( \hat{\gamma}_3 \) | \( |\hat{\gamma} - \hat{\gamma}|^2 \) |
|-----------------|-----------------|-----------------|-----------------|
| 1 | .250000 | 0 | .661438 | 2.354248 |
| 2 | .620324 | .361773 | .851451 | 1.594198 |
| 3 | .742057 | .245780 | .934356 | 1.262540 |
| 4 | .798431 | .274901 | .940106 | 1.239576 |
| 5 | .798799 | .274400 | .940331 | 1.238676 |
| 6 | .850590 | .301018 | .941841 | 1.232636 |
| 7 | .850871 | .300451 | .942067 | 1.231732 |
LITERATURE CITED


ACKNOWLEDGMENTS

The author is indebted to Professor H. O. Hartley for his valuable assistance in the preparation of this thesis and to both Professor Hartley and Professor C. G. Maple for their help in setting up a program of course work for a joint major in statistics and mathematics. Sincere thanks are expressed to Professor T. A. Bancroft for his advice and encouragement.

Acknowledgment is also made for the support of this work by a research grant from the National Science Foundation.
APPENDIX A. THE MODIFIED SIMPLEX METHOD
AND THE PRODUCT FORM OF THE INVERSE

In this section we give a brief discussion of some of the calculations involved in using the modified simplex method. The ordinary simplex method for solving the linear programming problem is well known and we shall not dwell on it here except to recall those parts of the computational procedure which are affected by the modification. For reference, we restate the linear programming problem as that of finding the vector \( x \) which maximizes the linear form

\[ c'x \]  

subject to the restrictions

\[ Ax \leq b \]
\[ x \geq 0 \]  

We let the \( m \times m \) matrix of basis vectors at the \( i^{th} \) step be denoted by \( B_i \) and let the row vector \( c_i' \) denote the corresponding elements from the objective function (169).

At the \( i^{th} \) stage of the ordinary simplex iteration all vectors in the tableau are expressed in terms of the present basis vectors. The selection of the next vector to come into the basis then proceeds as follows: The inner product of each of the columns in the tableau with the vector \( c_i' \) is
computed and denoted by, say, $z_j$ for the $j^{th}$ column. The corresponding entry in the objective function row, say $c_j$, is then subtracted and if $z_j - c_j$ is negative this column vector is eligible to come into the basis.

The modified simplex method was designed to overcome difficulties in error accumulation, to economize on the number of computations and to decrease the storage space necessary in the computer. As noted in Part I, it also plays an important role in the interpretation of the simplex calculations. In this modified method we do not compute the entire tableau at each stage, but instead we use the following method for selecting the next vector to come into the basis: If $P_j$ is a column vector in the initial tableau then its components in terms of the present basis are given by $B_i^{-1} P_j$. The inner product of this vector with $c_i'$, i.e., $c_i' B_i^{-1} P_j$, is the value for $z_j$. In practice, the objective function is written as the first row of the tableau and, in terms of partitioned vectors, the value of $z_j - c_j$ is given by

$$\begin{align*}
(1 \mid c_i' B_i^{-1}) \left(-\frac{c_j}{p_j}\right)
\end{align*}$$

(171)

The vector $(1 \mid c_i' B_i^{-1})$ is known as the "pricing vector" and the multiplication indicated in (171) is called the "pricing operation". As soon as a negative value for (171)
is obtained we designate the corresponding column to come into the basis.

To compute the matrix $B_i^{-1}$ necessary for the next iteration let us assume that the vector $P_j$ has been selected to come into the basis. Let us suppose that $P_j$ is to replace the vector in the $k^{th}$ column of the matrix $B_i$. In common terminology, $P_j$ is to "come in on row $k". Let the elements of the vector $B_i^{-1} P_j$ be denoted by $x_{lj}, l = 1 \ldots m,$ and form the matrix $E_i$ which is an $m \times m$ identity matrix except for the $k^{th}$ column which, written as a row vector, is

$$
\begin{pmatrix}
-\frac{x_{lj}}{x_{kj}} & \ldots & -\frac{x_{k-1, j}}{x_{kj}} & \frac{1}{x_{kj}} & -\frac{x_{k+1, j}}{x_{kj}} & \ldots & -\frac{x_{mj}}{x_{kj}}
\end{pmatrix}
$$

(172)

The matrix $B_i^{-1}$ is then given by the product $E_i B_i^{-1}$. As we proceed with the iterations, the inverse of the matrix of basis vectors is expressed as a product of matrices of the type $E_i$.

In practice, we use the $m+1 \times m+1$ matrix

$$
\begin{bmatrix}
1 & \begin{bmatrix} -c_i' \\
\vdots \\
0 \\
\end{bmatrix} \\
\begin{bmatrix} c_i \\
\vdots \\
0 \\
\end{bmatrix} & B_i
\end{bmatrix}^{-1} \begin{bmatrix}
1 & c_i' B_i^{-1} \\
\begin{bmatrix} -c_i \\
\vdots \\
0 \\
\end{bmatrix} \\
\begin{bmatrix} c_i \\
\vdots \\
0 \\
\end{bmatrix} & B_i^{-1}
\end{bmatrix}
$$

(173)

but the basic idea is still the same. In this case the
pricing vector is just the first row of the inverse matrix at each stage.

For further discussion of the modified simplex method we refer to Charnes and Cooper [7] and Orchard-Hays [25].
In this section we give in detail the calculations involved in applying the convex programming procedure of Part I. We adopt the following notation as in Appendix A:

(i) $B_i$ is the $3 \times 3$ matrix of basis vectors at the $i$th stage.

(ii) $c_i$ is the vector of values (with signs changed) from the first row of the tableau corresponding to the vectors in the basis $B_i$.

(iii) $c_i' B_i^{-1}$ is the current foot point.

(iv) $e_i$ is the vector selected to come into the basis and $\bar{e}_i$ is this same vector augmented by the element in the first row of the tableau.

The elements of the vectors are computed as indicated in Tableau 2 and the net prices are given by expressions (32), (33) and (34).

The initial basis $B_1$ consists of the vectors $S_1$, $S_2$ and $V$ shown in Tableau 2. Thus we have

$$B_1^{-1} = I_3$$

$$c_1' B_1^{-1} = (0 \ 0 \ -25) B_1^{-1} = (0 \ 0 \ -25)$$

With this foot point we compute the net prices for the sets
of restrictions A, B and C as:

(A) \(> 0\) \hspace{1cm} (B) -34 \hspace{1cm} (C) -116

Using the criterion discussed in Part I we select from set B and using the formulae given in Tableau 2 we get:

\[ \tilde{e}_1' = \begin{pmatrix} -34 \\ 12 \\ 1 \\ 0 \end{pmatrix} \]

To determine the vector to be removed from the basis we must compute \(B_1^{-1} P_0\) and \(B_1^{-1} e_1\) and consider the ratio of the corresponding elements of these vectors. The smallest non-negative ratio indicates the row on which \(e_1\) will come in, i.e., the column to be removed from the basis. In this case row 1 is selected and hence using the product form of the inverse we have:

\[
B_2^{-1} = E_1 B_1^{-1} = \begin{bmatrix} 1/12 & 0 & 0 \\ -1/12 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} B_1^{-1}
\]

The next foot point is:

\[ c_2' B_2^{-1} = \begin{pmatrix} 34 & 0 & -25 \end{pmatrix} B_2^{-1} = \begin{pmatrix} 2.833333 & 0 & -25 \end{pmatrix} \]

We record this vector with sign changed on the last element in Table 1 as the trial solution for step 2 and repeat the above process. Hereafter we shall only indicate the following
quantities which are relevant to the computation of the next trial solution:

(a) The net prices for sets A, B and C
(b) The restriction set we choose and the vector $\bar{e}_1$
(c) The non-unit vector $\eta$ of the matrix $E_1$ and the column in which it appears (see equation (172), Appendix A)
(d) The new foot point $c_{i+1}' B_{i+1}^{-1}$

**Step 3:**

(a) (A) > 0 (B) -8.028 (C) -51.36

(b) From set C

$$e_2' = \begin{pmatrix} -82.972224 & 14.333334 & 8 & 1 \end{pmatrix}$$

(c) In column 2

$$\eta' = \begin{pmatrix} -.175510 & .146939 & -.146939 \end{pmatrix}$$

(d) $c_3' B_3^{-1} = (34 \quad 82.97224 \quad -25) E_2 B_2^{-1} = (2.008486 \quad 9.897991 \quad -25)$

**Step 4:**

(a) (A) -4 (B) > 0 (C) -89

(b) From set A

$$e_3' = \begin{pmatrix} 6.540751 & .166195 & -1.237249 & 0 \end{pmatrix}$$
(c) In column 1
\[ \eta' = (4.283940, 0.78738, -0.787538) \]

(d) \[ c_4' B_4^{-1} = \begin{pmatrix} -6.540751 & 82.972224 & -25 \end{pmatrix} E_3 B_3^{-1} \]
\[ = \begin{pmatrix} 4.262679 & 5.859122 & -25 \end{pmatrix} \]

**Step 5:**

(a) (A) -1  (B) > 0  (C) -39

(b) From set C
\[ \tilde{e}_4' = \begin{pmatrix} -38.500257 & 11.474642 & -3.718244 & 1 \end{pmatrix} \]

(c) In column 1
\[ \eta' = (0.131389, -0.093588, -0.37801) \]

(d) \[ c_5' B_5^{-1} = \begin{pmatrix} 38.500257 & 82.972224 & -25 \end{pmatrix} E_4 B_4^{-1} \]
\[ = \begin{pmatrix} 6.268218 & 2.265812 & -25 \end{pmatrix} \]

**Step 6:**

(a) (A) > 0  (B) > 0  (C) -20

(b) From set C
\[ \tilde{e}_5' = \begin{pmatrix} -46.575539 & 7.463564 & 3.468376 & 1 \end{pmatrix} \]

(c) In column 2
\[ \eta' = (-0.147961, 2.147942, -0.999983) \]
\[(d) \quad c_6' B_6^{-1} = (38.500257 \quad 46.575539 \quad -25) \quad E_6 B_6^{-1} = (7.200321 \quad 5.142314 \quad -25)\]

**Step 7:**

(a) \[(A) - .5 \quad (B) > 0 \quad (C) -10\]

(b) From set \(A\)

\[\tilde{e}_6' = (3.398396 \mid -.122240 \quad -.642789 \quad 0)\]

(c) In column 1

\[\eta' = (15.445208 \quad 1.790378 \quad -.790378)\]

(d) \[c_7' B_7^{-1} = (-3.398396 \quad 46.575539 \quad -25) \quad E_6 B_6^{-1} = (7.824582 \quad 3.798986 \quad -25)\]

**Step 8:**

(a) \[(A) > 0 \quad (B) > 0 \quad (C) -5\]

(b) From set \(C\)

\[\tilde{e}_7' = (-15.343622 \mid 4.350836 \quad .402028 \quad 1)\]

(c) In column 1

\[\eta' = (.361757 \quad -.227261 \quad -.134494)\]

(d) \[c_8' B_8^{-1} = (15.343622 \quad 46.575539 \quad -25) \quad E_7 B_7^{-1} = (9.193822 \quad .852482 \quad -25)\]
Step 9:

(a) (A) > 0  (B) -7  (C) -17

(b) From set B
\[ \tilde{e}_g' = (50.526363 | -6.387644 1 0) \]

(c) In column 2
\[ \eta' = (4.280804 1.747357 -3.20788) \]

(d) \[ c_9' B_g^{-1} = (15.343622 -50.526363 -25) E_g B_g^{-1} \]
\[ = (8.766880 5.473178 -25) \]

Step 10:

(a) (A) -.5  (B) > 0  (C) -3.2

(b) From set A
\[ \tilde{e}_g' = (4.312744 | -.209271 -.684147) \]

(c) In column 3
\[ \eta' = (1.0 .631635 1.510889) \]

(d) \[ c_{10} B_{10}^{-1} = (15.343622 -50.526363 -4.312744) E_g B_g^{-1} \]
\[ = (8.490320, 3.706743 -23.086629) \]

Step 11:

(a) (A) -.197  (B) -.4949  (C) -.4517
(b) From set B
\[ \vec{e}_{10}' = (38.085534 \mid -4.980640 \ 1 \ 0) \]

(c) In column 2
\[ \eta' = (0 \ 1.266148 \ 0.389018) \]

(d) \[ c_{11}' B_{11}^{-1} = (15.343622 \ -38.085534 \ -4.312744) E_{10}B_{10}^{-1} \]
\[ = (8.396695 \ 3.735370 \ -22.690763) \]

Step 12:
(a) (A) -.1925 (B) -.008 (C) -.331

(b) From set A
\[ \vec{e}_{11}' = (3.136076 \mid -.188705 \ -.466921 \ 0) \]

(c) In column 3
\[ \eta' = (0 \ -.012483 \ 1.438493) \]

(d) \[ c_{12}' B_{12}^{-1} = (15.343622 \ -38.085534 \ -3.136076) E_{11}B_{11}^{-1} \]
\[ = (8.320138 \ 3.354005 \ -22.204272) \]

Step 13:
(a) (A) -.0093 (B) -.0290 (C) -.444

(b) From set B
\[ \vec{e}_{12}' = (35.224696 \mid -4.640276 \ 1 \ 0) \]
(c) In column 2

\[ \eta' = (0 \quad 1.067472 \quad .144504) \]

(d) \( c_{13}' B_{13}^{-1} = (15.343622 \ -35.224696 \ -3.136076) E_{12} B_{12}^{-1} \)

\[ = ( 8.314380 \quad 3.356328 \quad -22.180182) \]

Step 14:

(a) (A) -.009    (B) -.0003    (C) -.4358

(b) From set C

\[ e_{13}' = (-10.606148 \ | \ 3.371240 \quad 1.287344 \quad 1) \]

(c) In column 1

\[ \eta' = (1.0 \quad -.265121 \quad 1.328247) \]

(d) \( c_{14}' B_{14}^{-1} = (10.606148 \ -35.224696 \ -3.136076) E_{13} B_{13}^{-1} \)

\[ = ( 8.314380 \quad 3.356328 \quad -21.744357) \]

Step 15:

(a) (A) -.009    (B) -.00003   (C) -.00001

(b) From set A

\[ e_{14}' = (2.929862 \ | \ -.184113 \quad -.419541 \quad 0) \]

(c) In column 3

\[ \eta' = (0 \quad -.003190 \quad 1.105330) \]
(d) $c_{15}' B_{15}^{-1} = (10.606148 \ -35.224696 \ -2.929862) E_{14} B_{14}^{-1}$  
= ( 8.310157 \ 3.336657 \ -21.704780)

Step 16:

(a) (A) -.000195 (B) -.000168 (C) -.000373

(b) From set A

$\tilde{e}_{15}' = (2.919683 \ | \ -.183898 \ -.417082 \ 0)$

(c) In column 3

$\eta' = (0 \ -.000171 \ 1.005486)$

(d) $c_{16}' B_{16}^{-1} = (10.606148 \ -35.224696 \ -2.919683) E_{15} B_{15}^{-1}$

= ( 8.310081 \ 3.336274 \ -21.703978)

Net prices after final iteration:

(A) -.000021 (B) -.000200 (C) -.000336
APPENDIX C. SOME GEOMETRY OF SURFACES IN p-DIMENSIONS

In this section we review briefly the notions of the tangent plane and the normal line to a surface in p-dimensions. For the special case p = 3 we refer to Apostol [2, p. 325].

A surface in p-dimensions may be described as an aggregate of points whose coordinates are functions of p-1 variables. These p-1 variables are referred to as parameters or surface coordinates. To be consistent with the notation used in Part II, we assume that the surface has the following parametric representation

\[ x_i = i^f(v_2 \ldots v_p) \quad i = 1 \ldots p \]  

(174)

As the point \((v_2 \ldots v_p)\) is allowed to vary over some region in the p-1 dimensional \(v\)-space, the equations (174) define the surface in the p-dimensional \(x\)-space. Let us denote \(\frac{\partial i^f}{\partial v_j}\) by \(i^f_j\) and consider the p x p-1 matrix \(M\) of these partial derivatives:

\[
M = \begin{bmatrix}
i^f_2 & i^f_3 & \ldots & i^f_p \\
2^f_2 & 2^f_3 & \ldots & 2^f_p \\
\vdots & \vdots & \ddots & \vdots \\
p^f_2 & p^f_3 & \ldots & p^f_p
\end{bmatrix}
\]

(175)
We let \( iM \) denote the matrix obtained by deleting the \( i^{th} \) row of the matrix \( M \) and let \( i\tau \) be the determinant of \( iM \). The assumption that at least one of the \( i\tau \) is not identically zero is equivalent to the assumption of non-degeneracy of the surface defined by \((174)\). For example, in the case \( p = 3 \), if \( 1f, 2f \) and \( 3f \) are functions of \( t = \emptyset (v_2, v_3) \) then the equations \((174)\) define a curve not a surface. This can not happen if the rank of \( M \) is equal to \( p - 1 \) or equivalently that not all \( i\tau \) are identically zero. For our purpose we go one step further and insist that there be no surface point at which \( i\tau = 0 \), \( i = 1 \ldots p \). This restriction means that every point on the surface is a regular point, that is to say, there is a unique tangent plane at the point.

Since every point on the surface is a regular point, it follows by application of the implicit function theorem [see, e.g., Apostol, 2, p. 147] that we can represent the surface defined by \((174)\) implicitly by

\[
F(x) = 0
\]

\[(176)\]

**Lemma 1**: The vector \( \tau = (1\tau, -2\tau \ldots (-1)^{i-1}i\tau \ldots (-1)^{p-1}p\tau) \) evaluated at some surface point \((v_2 \ldots v_p)\) is the normal vector to the surface at that point.

**Proof**: We recall that a set of direction numbers for the normal vector at any point are given by
where \( F(x) \) is given by equation (176). Under the assumption that \( p^\tau \neq 0 \), we are able to express the surface defined by (174) or (176) in a third form as:

\[
x_p = g(x_1 \ldots x_{p-1})
\]

(178)

There is no loss of generality in assuming \( p^\tau \neq 0 \) at a particular point since by assumption at least one of the \( i^\tau \neq 0 \). We then have another set of direction numbers for the normal to the surface given by

\[
\frac{\partial x_p}{\partial x_1} : \frac{\partial x_p}{\partial x_2} : \ldots : \frac{\partial x_p}{\partial x_{p-1}} : -1
\]

(179)

Now by differentiating the equations (174) for \( i = 1 \ldots p-1 \) with respect to \( x_i \) we get the following system of equations:

\[
\frac{\partial f}{\partial x_i} = 1 = \sum_{j=2}^{p} k f_{ij} \frac{\partial v_j}{\partial x_i}
\]

(180)

\[
\frac{\partial f}{\partial x_i} = 0 = \sum_{j=2}^{p} k f_{ij} \frac{\partial v_j}{\partial x_i} \quad k = 1 \ldots p-1 \quad k \neq i
\]

In matrix form we express this system of equations as
We note that the coefficient matrix in equation (181) is simply $p^M$ and hence its determinant is $p^\tau$. Solving this system of equations we see that

$$
\frac{\partial v_{i+1}}{\partial x_i} = \frac{1}{p^\tau}
$$

$$
= (-1)^{i+j-1} \frac{p^\tau_{i,j}}{p^\tau}
$$

(182)
Here the determinant \( p^\tau_{ij} \) is simply the \( p-2 \times p-2 \) minor of \( p^\tau \) associated with the element \( i^f_j \). Now the partial derivatives in (179) may be expressed as \( \frac{\partial x^p}{\partial x^i} = \frac{1}{p^\tau} \sum_{j=2}^{p} p^f_j \frac{\partial v^j}{\partial x^i} \) which in view of equation (182) may be written as

\[
\frac{\partial x^p}{\partial x^i} = \frac{1}{p^\tau} \sum_{j=2}^{p} p^f_j (-1)^{i+j-1} p^\tau_{ij} \tag{183}
\]

It is easily seen that the determinant \( p^\tau_{ij} \), which is the minor of \( p^\tau \) associated with \( i^f_j \), is exactly equal to \( i^\tau_{pj} \), the minor of \( i^\tau \) associated with \( p^f_j \). We may thus write equation (183) in the more revealing form:

\[
\frac{\partial x^p}{\partial x^i} = \frac{(-1)^{i+1-p}}{p^\tau} \sum_{j=2}^{p} (-1)^{p+j-2} p^f_j i^\tau_{pj} \tag{184}
\]

The sum in equation (184) is just the expansion by minors of the determinant \( i^\tau \) about the last row, and we conclude finally that \( \frac{\partial x^p}{\partial x^i} = (-1)^{i+1-p} i^\tau/p^\tau \). Therefore, in terms of the determinants \( i^\tau \) we may rewrite the set of direction numbers given by equation (179) as

\[
(-1)^{2-p} \frac{i^\tau}{p^\tau} : (-1)^{3-p} \frac{2^\tau}{p^\tau} : \ldots : (-1)^{0} \frac{p-1^\tau}{p^\tau} : -l \tag{185}
\]

Since multiplication by a non-zero constant gives us another
set of direction numbers, we may multiply (185) by \((-1)^{2-p/p}\) to get the desired result of the lemma.

**Lemma 2:** The vector \(\tau\) of Lemma 1 is orthogonal to each of the columns vectors in the matrix \(M\).

**Proof:** Consider the inner product of the normal vector, \(\tau\), with the \(j^{th}\) column of \(M\), i.e., \(\sum_{i=1}^{p} (-1)^{i-1} f_{ij} i\tau\). The orthogonality is assured by the fact that this expression is merely the expansion of the determinant

\[
\begin{vmatrix}
1^{fj} & 1^{f2} & \cdots & 1^{fj} & \cdots & 1^{fp} \\
2^{fj} & 2^{f2} & \cdots & 2^{fj} & \cdots & 2^{fp} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
p^{fj} & p^{f2} & \cdots & p^{fj} & \cdots & p^{fp}
\end{vmatrix}
\]

(186)

about the first column. The fact that columns 1 and \(j\) are identical gives a zero value for the determinant.

Thus the vectors which are the columns of \(M\) determine a plane called the tangent plane to the surface and \(\tau\) is the orthogonal to this plane.

These results are merely a generalization of the familiar concepts for the case \(p=3\) [see Apostol, 2, p. 327]
in which case the two column vectors of $M$ determine the tangent plane and their vector product determines the normal vector.