NAVWEPS REPORT 9001
NOTS TP 3974

COMPUTING SCHEMES FOR FACTORIAL AND REGRESSION ANALYSIS AND DEVELOPMENT OF THE UNDERLYING THEORY USING LINEAR METHODS

by

S. R. Miller
and
D. R. Cruise

Propulsion Development Department

ABSTRACT. The basic concepts of regression and factorial analysis are reviewed and the theory developed, using the notation of linear algebra. The development includes the Gram-Schmidt construction, the linear method of least squares, and the use of orthogonal polynomials. Computing schemes applicable to digital computers are developed, and the basic coding is presented in the FORTRAN language. Fractional replication and the missing value problem are discussed.

U.S. NAVAL ORDNANCE TEST STATION
China Lake, California
November 1966

U. S. NAVAL ORDNANCE TEST STATION
AN ACTIVITY OF THE NAVAL MATERIAL COMMAND

J. I. HARDY, CAPT., USN
Commander

WM. B. MCLEAN, PH.D.
Technical Director

FOREWORD

This effort describes computing techniques used by the Analysis Branch for the analysis of experiments based on linear hypothesis statistics.

The text summarizes and provides insight into computer techniques developed during fiscal years 1957 through 1966 in support of environmental testing, the study of physical properties, static test, and many other technical areas of propulsion development as well as some administrative areas. The techniques are very general, and they may be applied to such diverse fields as rocket propulsion, agriculture, and medicine. The current Bureau of Naval Weapons Task Assignment is RMMP-22-028/216-1/F009-06-01.

The presentation has been reviewed for technical accuracy by Dr. D. E. Zilmer of the Mathematics Divison (Research Department) and by Stephen Lee and I. S. Kurotori of the Quality Assurance Division.

The material is presented for use at the working level and does not reflect the official view or the final judgment of the U.S. Naval Ordnance Test Station.

Released by
CRILL MAPLES, *Head*
Quality Assurance Division
11 October 1965

Under authority of
G. W. LEONARD, *Head*
Propulsion Development Department

NOTS Technical Publication 3974
NAVWEPS Report 9001

Published by .... Publishing Division
Technical Information Department
Collation .... Cover, 35 leaves, DD Form 1473, abstract cards
First printing .... 345 unnumbered copies
Security classification .... UNCLASSIFIED

CONTENTS

ACKNOWLEDGMENT

The special contributions and assistance of Randolph Eidemiller, Warren Oshel, and Stephen Lee are gratefully acknowledged.

INTRODUCTION

This report describes the continuation of two efforts. The first is a computing scheme developed by the authors for the analysis of factorial designs. An IBM 704 Electronic Data Processing Machine program using this scheme was submitted to the SHARE Society in 1958, and a short paper discussing this scheme was presented to the 14th national meeting of the Association for Computing Machinery (Ref. 1).

The second effort is an attempt to promote a general linear approach to numerical and statistical analysis. A report describing this approach in the smoothing of experimental data has already been published (Ref. 2).

Three concepts form the basis of these efforts: the Gram-Schmidt construction, the linear method of least squares, and the application of orthogonal polynomials. References 3, 4, and 5, and many other texts, also introduce these well-known concepts.

The notation of vectors, matrices, and tensors is used in place of the summation notation frequently (but not universally) used in textbooks. It is believed that the notation used is more practical at the working level, with the advent of FORTRAN and ALGOL, and it certainly allows a much shorter text.

Computer programs for factorial and statistical analysis are presented in Appendixes A, B, and C; they demonstrate a method rather than finished programs. They are finished to the extent that results are summarized, but there are many ways of summarizing, depending principally on the desires of the user. Because of the versatility of the FORTRAN language, the prospective user is invited to summarize results in the manner he sees fit.

This work is presented principally to those who are computer-oriented. For this reason, there is a great deal of elaboration on some of the basic concepts and philosophy of statistics. On the other hand, it is assumed that the reader is familiar with the algebra of vectors and matrices and their geometric interpretations.

VECTORS, MATRICES, AND THE GRAM-SCHMIDT CONSTRUCTION

Linear hypothesis statistics, which includes regression and factorial analyses, may be developed by a method (the orthogonalization of matrices), a theorem (the Pythagorean), and a philosophy (inference). This development is best made with a uniform notation. Therefore, the following definitions are made.

Matrices will be denoted by capital letters such as X, Y, A, and D, and *scalar* quantities will be presented by small letters such as a, b, and c.

The *elements* of a matrix will be represented by subscripted small letters such as x_{ij} and y_i.

The ith *row* of a matrix X will be denoted by $x_{i.}$, and the jth *column* of a matrix X will be denoted by $x_{.j}$. A row or a column (or any singly subscripted variable) may be considered to be a *vector* in space, the elements of the row or column being the ordered components of the vector. This is the geometric viewpoint.

Two columns of a matrix are said to be *orthogonal* if

$$x_{.j} \cdot x_{.k} = 0 \qquad j \neq k \tag{1}$$

where the *vector dot product* notation above is defined thus

$$x_{.j} \cdot x_{.k} \equiv \sum_{i=1}^{n} (x_{ij})\,(x_{ik}) \tag{2}$$

From the geometric viewpoint, the vectors $x_{.j}$ and $x_{.k}$ meet at right angles in n-dimensional space.

An individual column of a matrix X is said to be *normalized* if

$$x_{.j} \cdot x_{.j} = 1 \tag{3}$$

An unnormalized column (provided it contains at least one finite entry) may be easily normalized by dividing each element of the column by the length of the column vector as shown:

$$x'_{ij} = x_{ij}/|x_{.j}| \qquad 1 \leq i \leq n$$

and where the length is defined as follows:

$$|x_{.j}| = \sqrt{x_{.j} \cdot x_{.j}} \equiv \sqrt{\sum_{i=1}^{n} x_{ij}^{\,2}} \tag{4}$$

From the geometric viewpoint, a normalized column is a vector of unit length. The vector need not be oriented along any axis but may have components along all axes. Equation 4 rewritten

$$|x_{.j}|^2 = x_{.j} \cdot x_{.j} \equiv \sum_{i=1}^{n} x_{ij}^{\,2}$$

is a statement of the Pythagorean theorem generalized to m dimensions. The square of the hypotenuse (vector length) is equal to the sum of the squares of the other sides (components).

A matrix X is said to be *orthonormal* (with respect to columns) if each column is normalized and if each column is orthogonal to every other column. From a basic theorem of linear algebra, this is possible only if the number of columns does not exceed the number of rows (Ref. 3).

An important property of any orthonormal matrix is that premultiplication by its own transpose[1] yields an identity matrix.

$$P'P = I \tag{5}$$

This may be demonstrated as follows:

$$\begin{bmatrix} \sqrt{\frac{1}{3}} & \sqrt{\frac{1}{3}} & \sqrt{\frac{1}{3}} \\ -\sqrt{\frac{1}{2}} & 0 & \sqrt{\frac{1}{2}} \end{bmatrix} \begin{bmatrix} \sqrt{\frac{1}{3}} & -\sqrt{\frac{1}{2}} \\ \sqrt{\frac{1}{3}} & 0 \\ \sqrt{\frac{1}{3}} & +\sqrt{\frac{1}{2}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Note that it is the second matrix which is orthonormal with respect to columns.

[1] The same matrix with rows and columns interchanged

Conversely, the product $P\,P'$ does *not* generally yield an identity matrix unless P is square. This also may be demonstrated

$$\begin{bmatrix} \sqrt{\frac{1}{3}} & -\sqrt{\frac{1}{2}} \\ \sqrt{\frac{1}{3}} & 0 \\ \sqrt{\frac{1}{3}} & +\sqrt{\frac{1}{2}} \end{bmatrix} \begin{bmatrix} \sqrt{\frac{1}{3}} & \sqrt{\frac{1}{3}} & \sqrt{\frac{1}{3}} \\ -\sqrt{\frac{1}{2}} & 0 & \sqrt{\frac{1}{2}} \end{bmatrix} = \begin{bmatrix} \frac{5}{6} & \frac{1}{3} & -\frac{1}{6} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ -\frac{1}{6} & \frac{1}{3} & \frac{5}{6} \end{bmatrix}$$

A matrix X may be transformed into an orthonormal matrix P by the Gram-Schmidt construction (Ref. 3).[2] The following paragraphs describe this construction. The matrix is first transformed into a matrix having the orthogonal property

$$X \longrightarrow Q$$

Thence it is transformed into a matrix which also has the normal property

$$Q \longrightarrow P$$

The first column of the orthogonal Q matrix is formed by simply using the first column of the X matrix[3]

$$q_{.1} = x_{.1} \tag{6}$$

The second column of the Q matrix is formed by taking the second column of the X matrix and adding to it that portion, c_{12}, of the first column of the Q matrix which makes the new column orthogonal to the first column of the Q matrix. The following vector equation expresses this required condition

$$q_{.1} \cdot (x_{.2} + c_{12}\, q_{.1}) = 0 \tag{7}$$

The solution yields

$$c_{12} = \frac{-q_{.1} \cdot x_{.2}}{q_{.1} \cdot q_{.1}}$$

With this result, the second column of the Q matrix is

$$q_{.2} = x_{.2} + c_{12}\, q_{.1} \tag{8}$$

The third column of the Q matrix is formed by taking $x_{.3}$ and adding portions c_{13} and c_{23} of the $q_{.1}$ and $q_{.2}$ columns which make $q_{.3}$ orthogonal to both $q_{.1}$ and $q_{.2}$. Two simultaneous vector equations express these desired conditions

$$q_{.1} \cdot (x_{.3} + c_{13}\, q_{.1} + c_{23}\, q_{.2}) = 0 \tag{9}$$

$$q_{.2} \cdot (x_{.3} + c_{13}\, q_{.1} + c_{23}\, q_{.2}) = 0 \tag{10}$$

[2] Provided the columns have the property of linear independence. A further elaboration of this is given after the mechanics of the construction is presented.

[3] Remembering that the dot notation is defined thus

$x_{.1} = x_{i1} \qquad 1 \leq i \leq n$

Since $q_{.1} \cdot q_{.2} = 0.$, c_{13} and c_{23} are quickly evaluated to be

$$c_{13} = \frac{-q_{.1} \cdot x_{.3}}{q_{.1} \cdot q_{.1}} \tag{11}$$

$$c_{23} = \frac{-q_{.2} \cdot x_{.3}}{q_{.2} \cdot q_{.2}} \tag{12}$$

With these results, the third column of the orthogonal matrix is

$$q_{.3} = x_{.3} + c_{13}\, q_{.1} + c_{23}\, q_{.2} \tag{13}$$

The Gram-Schmidt construction may be easily generalized so that the expression for the *j*th column, $q_{.j}$, may be written

$$q_{.j} = x_{.j} + \sum_{k=1}^{j-1} q_{.k}\, c_{kj} \qquad 1 \leq j \leq m \tag{14}$$

This may be expressed by the matrix relation

$$Q = X + QC \tag{15}$$

where

$$\left.\begin{array}{ll} c_{kj} = 0 & j \leq k \\ c_{kj} = \dfrac{-q_{.k} \cdot x_{.j}}{q_{.k} \cdot q_{.k}} & j > k \end{array}\right\} \begin{array}{l} 1 \leq j \leq m \\ 1 \leq k \leq m \end{array}$$

and where *m* is the number of columns and *n* is the number of rows of the *X* and *Q* matrices.

The orthogonal matrix *Q* may now be transformed into the orthonormal matrix *P* by dividing each element q_{ij} by the length of the column vector to which it belongs

$$p_{ij} = |q_{.j}|^{-1}\, q_{ij} \qquad \begin{array}{l} 1 \leq i \leq n \\ 1 \leq j \leq m \end{array}$$

This too may be expressed by a matrix relation

$$P = QR \tag{16}$$

where

$$\left.\begin{array}{ll} r_{jk} = 0 & j \neq k \\ r_{jk} = |q_{.j}|^{-1} & j = k \end{array}\right\} \begin{array}{l} 1 \leq j \leq m \\ 1 \leq k \leq m \end{array}$$

Normalization may not be possible. This is obviously the case where all elements of a column are zero. When such a column vector appears in a matrix during Gram-Schmidt construction, the original matrix lacks the required property of linear independence. In fact, this is an excellent practical test of linear independence.

Two important relations have been developed above

$$Q = X + QC \tag{15}$$

$$P = QR \tag{16}$$

The first may be rearranged as follows:

$$Q\,(I - C) = X \tag{17}$$

$(I - C)$ is a triangular matrix with diagonal terms of unity; its determinant is nonzero (in fact, unity) and, therefore, its inverse existe, and Eq. 17 may be rewritten

$$Q = X\,(I - C)^{-1} \tag{18}$$

From Eq. 16, P is found to be

$$P = X\,(I - C)^{-1} R \tag{19}$$

or $\quad P = XD$

where $\quad D = (I - C)^{-1} R$

Equation 19 demonstrates that the Gram-Schmidt construction is equivalent to postmultiplication of the original matrix by another matrix, D.

There is also a geometric viewpoint of the construction. The columns of the X matrix represent vectors of varying lengths, meeting each other at various angles in n-dimensional space. Each vector is considered in the arbitrary order in which they appear in the X matrix, and its components which lie along previously chosen vectors are removed. The first vector obviously has no other vector to be compared with and is left alone.

This can be illustrated graphically in the two-dimensional case (Fig. 1). Consider two vectors, $\vec{x}_{.1}$ and $\vec{x}_{.2}$, described by a matrix X, where x_{11} is the component of $\vec{x}_{.1}$ on the x_1 axis,

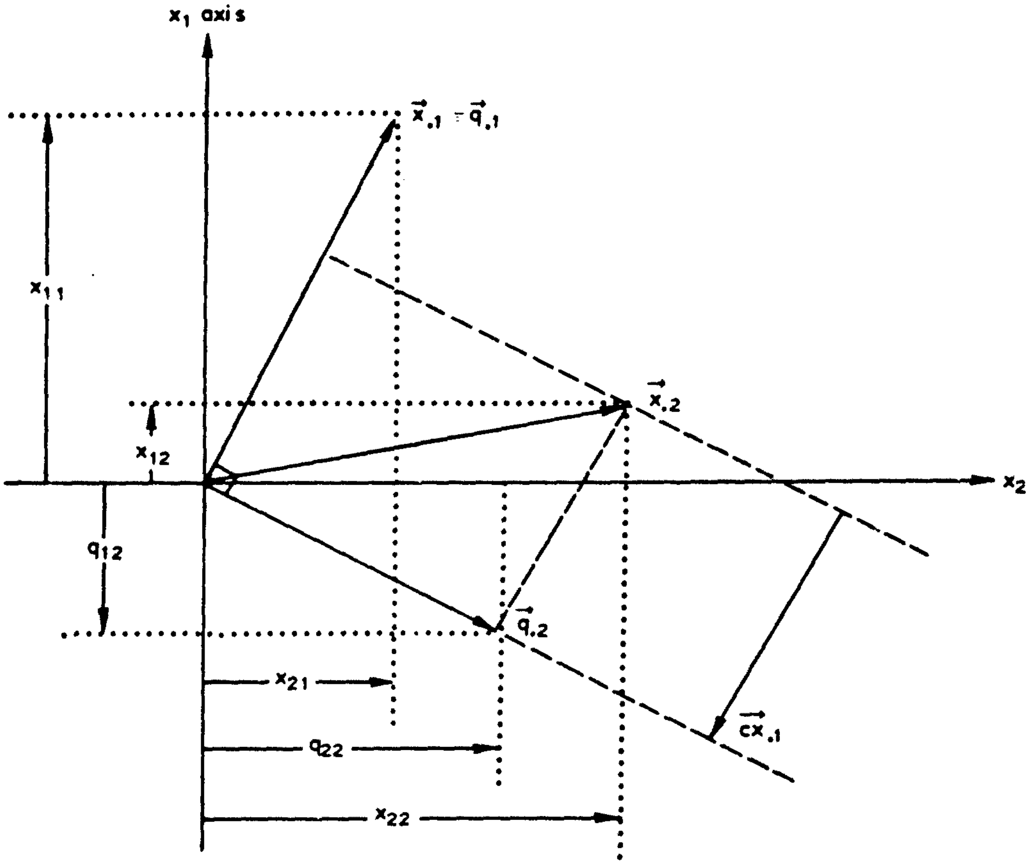

FIG. 1 Geometric Representation of the X and Q Matrices in Two Dimensions, Showing the Gram-Schmidt Construction.

and x_{21} is the component of $\vec{x}_{.1}$ on the x_2 axis, x_{12} is the component of $\vec{x}_{.2}$ on the x_1 axis, and x_{22} is the component of $\vec{x}_{.2}$ on the x_2 axis. In the Gram-Schmidt construction the column $x_{.1}$ is accepted, and its elements are put into the first column of the matrix Q.

Then the vector $\vec{x}_{.2}$ is taken, and its component $(\overline{c\dot{x}}_{.1})$ along $\dot{x}_{.1}$ is subtracted to form the vector $\vec{q}_{.2}$. The components of $\vec{q}_{.2}$ along the x_1 and x_2 axes now form the second column of an orthogonal matrix Q. The normalizing process (not shown in Fig. 1) shrinks $\vec{q}_{.1}$ and $\vec{q}_{.2}$ to unit lengths but preserves their respective directions, which now form a right angle with each other.

On digital computers, the matrices P and D are determined from X as follows: The P matrix is formed in the same memory area as the X matrix. Each succeeding column vector of the X matrix (except the first) is orthogonalized separately with respect to each succeeding column in the manner previously described. The separate columns of the matrix may then be normalized. Meanwhile, the same operations performed on X are also performed in another memory area originally containing an identity matrix I. It is obvious that as X is transformed into P, I is transformed into D, thus

$$X \longrightarrow P$$
$$I \longrightarrow D$$

because, as noted above, this construction is equivalent to postmultiplication by the matrix D.

A numerical example of these transformations appears in Appendix D.

REGRESSION ANALYSIS AND THE LINEAR METHOD OF LEAST SQUARES

The linear method of least squares is closely related to the solution of simultaneous linear equations by the method of orthogonalization which will be briefly described.

The following system of linear equations,

$$\left.\begin{array}{c} a_1 x_{11} + a_2 x_{12} + \ldots + a_n x_{1n} = y_1 \\ a_1 x_{21} + a_2 x_{22} + \ldots + a_n x_{2n} = y_2 \\ \ldots \\ a_1 x_{n1} + a_2 x_{n2} + \ldots + a_n x_{nn} = y_n \end{array}\right\} \tag{20}$$

may be represented by the matrix equation

$$XA = Y \tag{21}$$

where the X and Y matrices are known, and A is to be determined.

With the results of the first section, a solution may be found as follows:

$$P\,D^{-1}\,A = Y \qquad \text{(from Eq. 19)} \tag{22}$$

$$P'\,P\,D^{-1}\,A = P'\,Y \tag{23}$$

$$D^{-1}\,A = P'\,Y \qquad \text{(from Eq. 5)} \tag{24}$$

hence

$$A = D P' Y \tag{25}$$

A FORTRAN coded subroutine for simultaneous linear equations using this method appears in Ref. 2.

At this point, it is useful to look at the geometric viewpoint of the system of linear equations and its solution as outlined above. Y, of course, should be considered a vector in n-dimensional Euclidean space, with a direction, a length, and components y_i. As before, the dot product of Y on Y should be recognized as an expression of the Pythagorean theorem:

$$Y \cdot Y = |Y|^2 = \sum y_i^2 \tag{26}$$

where $|Y|$ is the length of the vector. The y_i are components of Y when referred to a particular reference frame, namely, that defined by the identity matrix I of order n.

Note that Eq. 25 may be rewritten $B = P' Y$, where $B = D^{-1}A$. Since P is orthonormal, the elements of B are also components of Y.[4] This time the components b_i are along unit vectors whose components, in turn, are found in the columns of P, and hence there is a new reference frame P. The Pythagorean theorem may also be stated in terms of this reference frame

$$|Y|^2 = \sum b_i^2 = |B|^2 \tag{27}$$

Inherent in this development is the fact that an orthonormal matrix is a *vector basis*, or Cartesian reference frame, in which the Y vector retains both its magnitude and direction.[5] In other words, the vector B in the P reference frame is identical with the Y vector in the I reference frame.

The solution process can be described in this light. Consider the matrix equation $X A = Y$. A may be called a representation of Y, because it may be determined from Y. Even more, it represents Y in a manner very pertinent to X, so it may be stated further that A is a representation of Y in the X space. But it is a distorted representation of Y, because the X space is generally distorted.[6] If this distortion is removed by Gram-Schmidt construction, there is obtained a space, P, in which Y has an undistorted, easily calculated, understandable, representation, B. But B is not what is wanted, so, to get A, one must reintroduce the distortion. This is done in the final step of the solution $A = DB$ (from Eq. 25). Thus the term "distortion," used very loosely above, is, nevertheless, precisely and quantitatively expressed by the matrix D.

To turn now from the geometric viewpoint, consider the case where the columns of the X matrix do not possess linear independence. This is revealed when a column of zeros appears during the Gram-Schmidt construction. When this happens, the vector basis is incomplete, because there are fewer vectors than dimensions. The vector basis is also incomplete, for the same reason, when there are fewer columns than rows in X. This will be considered to be the general case of the incomplete vector basis, because zero columns which appear during Gram-Schmidt construction may be simply deleted with no loss of generality.

It will first be shown why this situation (fewer columns than rows) actually occurs. Suppose one is conducting a series of experiments to measure the value of y, the dependent variable. Under experimental control are m independent variables x_j. By physical law or statistical hypothesis, y may vary linearly with the x_j as follows:

$$y = a_1 x_1 + a_2 x_2 + \ldots + a_m x_m \tag{28}$$

Equation 28 is called the "regression model."

[4] Reference 3, or another basic text, may be consulted to prove this point

[5] It is simply two different methods of locating its coordinate points

[6] Not orthonormal

Obviously, one must perform at least m experiments to determine the a_j. However, one may proceed further and perform n experiments, where n is any number greater than m. Attaching the subscript i for the ith experiment, one gets the following set of "overdetermined" equations

$$a_1 x_{i1} + a_2 x_{i2} + \dots + a_m x_{im} = y_i \qquad 1 \leq i \leq n \tag{29}$$

which may also be represented by the matrix equation

$$XA = Y \tag{21}$$

noting that X now contains more rows than columns.

While it is not surprising that the experimenter did not have to stop after m experiments, it may be surprising that the overdetermined[7] equations above may be solved in the same manner as outlined in Eq. 21 through 25. A careful examination will show that no step is invalid.

The preceding is contrary to usual experience in solving systems of linear equations. If the X matrix is not linearly independent, no solution is possible by the usual methods. But here there is a solution where the X matrix is not even square, let alone linearly independent. Before examining the meaning of such a solution, a numerical illustration will be given.

Let the dependent variable be the yield of the principal product in a chemical experiment. One is interested in how the yield varies with pressure (x_2), temperature (x_3), and amount of catalyst (x_4). Six experiments provide the following data:

Experiment No. (i)	Pressure (x_{i2})	Temperature (x_{i3})	Catalyst (x_{i4})	Yield (y_i)
1	1	300	.001	108
2	1	400	.001	111
3	1	600	.002	120
4	2	350	.001	99
5	2	550	.002	115
6	2	650	.002	117

The following regression model can then be set up:

$$a_1 x_{i1} + a_2 x_{i2} + a_3 x_{i3} + a_4 x_{i4} = y_i \qquad 1 \leq i \leq 6 \tag{30}$$

In this case (which is very common), a dummy x_1 variable must be defined which takes the value unity for all values of i. The data and model are now expressed in the following matrix:

$$\begin{bmatrix} 1 & 1 & 300 & .001 \\ 1 & 1 & 400 & .001 \\ 1 & 1 & 600 & .002 \\ 1 & 2 & 350 & .001 \\ 1 & 2 & 550 & .002 \\ 1 & 2 & 650 & .002 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} 108 \\ 111 \\ 120 \\ 99 \\ 115 \\ 117 \end{bmatrix} \tag{31}$$

P and D matrices are now found by Gram-Schmidt construction, and the A matrix is found by Eq. 25 to be (99.417, -7.250, .025, 7500).

What is the A vector in this case? The A vector is an incomplete representation of the Y vector in the X space. It has only four components: to be complete, it should have six.[8]

Further insight may be gained by returning to the geometric viewpoint and to the solution process itself. The Gram-Schmidt construction has developed a matrix P, which contains four

[7] More rows than columns in the X matrix; equivalently more data than parameters in an experiment.

[8] In statistical terminology, components are called degrees of freedom.

columns in the example. These represent four vectors meeting at right angles in six-dimensional space. Therefore, when the solution step $B - P'Y$ is performed, only four of the six possible components, b_i, are computed. It would be possible to compute two more components by arbitrarily adding two more orthogonal vectors to the P space. However, these two new vectors would have no definite relation to the independent variables of the experiment, so this addition would not be justified. One must accept the actual situation.

B, less two components, while not a complete representation of Y in the P space, is a complete representation of some other vector, $\hat{Y}$, which does not have components along the two missing P vectors. $\hat{Y}$ can be found by the reverse transformation $\hat{Y} = PB$, which transforms B back onto the I reference frame. (This expression may be shown to be equivalent to $\hat{Y} = XA$, where A is obtained from Eq. 25.)

In the example, $\hat{Y}$ and Y compare as follows:

$\hat{Y}$	Y
107.2	108
109.7	111
122.2	120
101.2	99
113.7	115
116.2	117

It is not an exact solution, but it is close, and there is good reason why it should be close. If the physical law is correct and there are no experimental errors, one would expect the Y vector to have only four components in its A representation, namely, those of the regression model. The manifestation of other components, as seen in the difference between Y and $\hat{Y}$, would be expected only if there were experimental error or if the model were incorrect. Thus, it is not only possible to have an overdetermined set of equations, but it can be very desirable, because it can elucidate sources of error. It will now be shown that this is not only *a* solution to the overdetermined set of linear equations, but that it is the *best* solution in the sense of least squares.

The problem of least squares can be stated from the geometric viewpoint:

The best representation[9] of a vector Y of dimension n, in a subspace P of dimension m, smaller than n, is found by constructing the vector $\hat{Y}$, having components only in P, such that the square of the distance between Y and $\hat{Y}$ is a minimum.

The distance between two vectors is the length of the vector difference, denote $|Y - \hat{Y}|$. The evaluation of this quantity follows. The unknown vector $\hat{Y}$ has components $\hat{b}_1, \hat{b}_2, \ldots, \hat{b}_m$ in the P space, which are chosen to fulfill the criterion above. If additional vectors are added to P (in practice, a trial and error process) so as to form a complete orthonormal vector basis, the components of the known vector Y may be exactly determined,[10] $b_1, b_2, \ldots, b_m, b_{m+1}, \ldots, b_n$, where the last $n - m$ components do not belong to the original incomplete space. The components of the vector difference $Y - \hat{Y}$ are found by the rule of vector subtraction

$$(b_1 - \hat{b}_1), (b_2 - \hat{b}_2), \ldots, (b_m - \hat{b}_m), b_{m+1}, \ldots, b_n$$

The square of the vector difference (which is to be minimized) is found by the theorem of Pythagoras

$$|Y - \hat{Y}|^2 = (b_1 - \hat{b}_1)^2 + (b_2 - \hat{b}_2)^2 + \ldots (b_m - \hat{b}_m)^2 + (b_{m+1})^2 + \ldots + b_n^2 \qquad (32)$$

[9] In the sense of least squares.

[10] As before, $B = P'Y$.

The minimum occurs when each $\hat{b}_i$ is set equal to b_i for all i up to m; this can be seen by observing that each of the leading terms has a minimum possible value of zero. Thus, the criterion of least squares is fulfilled when $\hat{Y}$ consists of the components of Y in the incomplete P space. This is exactly what the solution process above does, and it will henceforth be referred to as the linear method of least squares.

The foregoing is not the classical approach to least squares, nor does it yield the more familiar least squares equations. It is a curiosity in statistics that, while much of the theory is based on linear hypotheses, much of the analysis is presented in a cumbersome quadratic form. The subject of least squares is easily presented with the aid of the Pythagorean theorem, as shown above. In the classical approach, differential calculus and summation algebra are used. To arrive at the quadratic form of the least squares equations, modify the linear equation $XA = Y$ by premultiplying both sides by X', which yields $(X'X)A = (X'Y)$, which, expanded, yields

$$\begin{bmatrix} \sum_{i=1}^{n} x_{i1}x_{i1} & \sum_{i=1}^{n} x_{i1}x_{i2} & \cdots & \sum_{i=1}^{n} x_{i1}x_{im} \\ \sum_{i=1}^{n} x_{i2}x_{i1} & \sum_{i=1}^{n} x_{i2}x_{i2} & \cdots & \sum_{i=1}^{n} x_{i2}x_{im} \\ & \cdots & & \\ \sum_{i=1}^{n} x_{im}x_{i1} & \sum_{i=1}^{n} x_{im}x_{i2} & & \sum_{i=1}^{n} x_{im}x_{im} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \cdots \\ a_m \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} x_{i1}y_i \\ \sum_{i=1}^{n} x_{i2}y_i \\ \cdots \\ \sum_{i=1}^{n} x_{im}y_i \end{bmatrix} \qquad (33)$$

This is the form presented in many texts.

A sample coding for regression analysis, using the linear method, appears in Appendix A. Statisticians use the inverse matrix elements from the quadratic form to perform t tests, which show whether the data conform to the regression model or not. The reader is referred to standard works (such as Ref. 1) for a description of these tests. The statistician is not deprived of these elements in the linear method, for one can show that the inverse matrix $(X'X)^{-1}$ is equal to the product DD' from the linear method. The sample coding in Appendix A includes this calculation.

ORTHOGONAL POLYNOMIALS AND CURVILINEAR REGRESSION

In the previous secti. r-s it was shown that the square matrix X of a matrix equation may be transformed into a matrix P which has highly desirable algebraic and more readily understood geometric properties. The column vectors of the P matrix are of unit length and are mutually perpendicular to one another. Furthermore, the set of vectors may be incomplete, in which case there are fewer vectors than data points.

The elements of the orthogonal vectors are called by statisticians "orthogonal polynomials," for the vector elements are actually values taken by the set of polynomial functions

$$\left.\begin{aligned} p_{i1} &= d_{11} x_{i1} \\ p_{i2} &= d_{12} x_{i1} + d_{22} x_{i2} \\ p_{i3} &= d_{13} x_{i1} + d_{23} x_{i2} + d_{33} x_{i3} \\ &\ldots \end{aligned}\right\} \quad 1 \leq i \leq n \qquad (34)$$

These equations are an expansion of the matrix product, $P=XD$, from the preceding section.

The most common application of orthogonal polynomials occurs in curvilinear regression. Curvilinear regression is a special case of regression analysis based on the following model

$$y_i = a_1 + a_2 x_i + a_3 x_i^2 + \ldots a_m x_i^{m+1} \quad 1 \leq i \leq n \qquad (35)$$

Again, this may be represented by a matrix equation

$$XA = Y \qquad (21)$$

where the ijth element of X is defined by

$$x_{ij} = x_i^{j-1} \qquad (36)$$

and where the singly subscripted x_i are the values taken by a single independent variable, and $j-1$ is an exponent. As before, Gram-Schmidt construction yields a P matrix which is related to X as follows:

$$\begin{aligned} p_{i1} &= d_{11} \\ p_{i2} &= d_{12} + d_{22} x_i \\ p_{i3} &= d_{13} + d_{23} x_i + d_{33} x_i^2 \\ &\ldots \end{aligned} \qquad (37)$$

In the functions above, p_{i1} is a constant, p_{i2} is a linear function of x_i, p_{i3} is a quadratic function of x_i, and so on. A very important facet of these equations should be noted here. Although the members on the right are polynominal functions (of linear, quadratic, and higher degree varieties), the dependent left-hand members (being only the set of functional values) may by viewed as vectors and represented in a linear vector space. This will be an aid to understanding how one can speak of higher order variations with a linear theory.

Continuing to the next step of the linear method

$$B = P'Y \qquad (38)$$

This equation may be broken down as follows:

$$\begin{aligned} b_1 &= P_{.1} \cdot Y_. \\ b_2 &= P_{.2} \cdot Y_. \\ b_3 &= P_{.3} \cdot Y_. \\ &\ldots \end{aligned} \qquad (39)$$

The significance of the b_j, as the components of Y in the P reference frame, has already been discussed. In addition to the algebraic and geometric meaning, the b_j will be shown to take on another meaning. It is about the b_j components of Y that the statistician begins making inferences. It is inference that separates the statistics from the physics and mathematics of the experiment and of its model. Equation 39 will serve as an illustration. The term b_2 may be inferred to be the variation of y_i due to a linear variation of x_i; b_3 may be inferred to be due to the quadratic variation, and so on through higher orders.

That inference is not strictly mathematical may be seen by the word "due." There is almost a causal implication here, although statistics are not supposed to show causes![11] What is meant is that the b_j (which, mathematically, are only linear vector components) are inferred to show a true functional relation (not necessarily linear) between the dependent and independent variables. Under certain frequent conditions,[12] it is possible to show that the b_j are indeed the best estimates of the true functional relations, in which case the statistician has great confidence in his inferences.

Continuing with the linear method for the case at hand (curvilinear), the last step is to transform the B vector into the A vector, by premultiplying with the D matrix, and thus complete the determination of the regression coefficients. This last step may be considered to be an anticlimax. Before the last transformation, there existed an ideal Euclidean[13] space, where the b_j were merely components of Y in the P reference frame. The last transformation not only destroys this situation but also destroys the inferences as well, because the magnitude of the a_i depend not only on the linear, quadratic, cubic, etc., variations of y with x, but also on the amount of distortion in the X matrix.

If the last transformation is deferred, there are not only components about which inferences can be made, but there is also a simple device at hand to test the validity of the model. In the preceding section, it was shown that Y may have components which do not appear in the P reference frame. The hypothesis may be made that these components are experimental error. If the error is small, the components of Y about which functional relations of y with x were inferred should be larger than those about which functional relations were not inferred. The following comparison function may therefore be set up

$$F_j = \frac{b_j^2}{\frac{1}{n-m} \sum_{k=m+1}^{n} b_k^2} \qquad 1 \leq j \leq m \tag{40}$$

which is called an F ratio. Note the b_k in the denominator are those components of Y which do not appear in the P space.

An F distribution is a function of the number of inference components in the numerator and the number of error components in the denominator. When an F is larger than a preset percentage (say 95%) of all other possible Fs that are determined by the distribution, the statistician says that a true functional relation exists. In normal practice, the computer calculates the F ratios, and the statistician draws inferences and performs the F tests. F and t tests are part of the probability aspects of statistics, and it is not in the scope of this report to pursue them further. The reader is referred to Ref. 4 and other basic works on statistics.

[11] For an interesting discussion of this, see How To Lie With Statistics, by D. Huff, W. W. Norton and Co., New York, 1954. 142 pp.

[12] Namely, normal (Gaussian) distribution of error.

[13] That is, undistorted, orthonormal.

The computational aspects of curvilinear regression will now be considered. It will be shown that the calculations of the linear method can be performed stepwise. As the name implies, stepwise calculation allows the regression coefficients to be calculated and tested one at a time. Such a process is desirable because the curvilinear regression model approximates a Taylor's expansion of y on x

$$y_i = c_1 + a_2 x + a_3 x^2 + \dots + a_n x^{n-1} \tag{41}$$

where

$$a_i \cong \left[\left(\frac{d}{dx}\right)^{i-1} y\right]_0 \frac{1}{(i-1)!} \tag{42}$$

Because the reciprocal factorial term in a_i rapidly approaches exceedingly small numbers, it is expected that at some term in the model, the remaining functional relation between x and y is small compared to the experimental error. A stepwise solution can locate this term and can, furthermore, yield regression coefficients for all intermediate degree models without numerical repetition.

Having established the motive for using stepwise regression, it will be shown how it works for digital computers. One possible method is to use published values of orthogonal polynomials, but this is not usually practical on digital computers. Only the orthogonal polynomials for equally spaced intervals of the independent variable are usually published. Also, the D matrix is very hard to determine, requiring almost as much computation as the Gram-Schmidt construction. A more practical approach to the use of digital computers will be described.

Stepwise regression is the technique of performing all steps of the linear method by analyzing one column of the X matrix at a time, according to the following outline, which describes the steps for the jth column

1. Form the jth column of the X matrix

$$x_{ij} = x_i^{j-1} \qquad 1 \leq i \leq n \tag{43}$$

2. Orthogonalize this column with respect to previous columns by the Gram-Schmidt construction, and normalize to find the jth column of the P and D matrices

$$p_{ij}, d_{ij} \qquad 1 \leq i \leq n \tag{44}$$

3. Find the jth component of Y in the P space[14]

$$b_j = p_{.j} \cdot y. \tag{45}$$

4. Compute the current F ratio

$$F(1, n-j) \quad \frac{b_j^{\,2}}{\frac{1}{n-j}\sum_{k=j+1}^{n} b_k^{\,2}} \tag{46}$$

5. The F ratio of step 4 may be used to test for significance of the jth functional relation. At any time before column m is reached, the statistician may decide to stop the stepwise regression on the basis of this test, or perhaps on the basis of the residual standard deviation discussed below. It is difficult to get agreement on a proper criterion.

[14] The vector $y.$ is identical to the Y matrix

6. At any time, the conventional regression coefficients for a j-1 degree fit can be found from the current states of the D and B matrices

$$a_i = d_{ij} \cdot b_j \tag{47}$$

This technique is used in the sample program shown in Appendix B.

The denominator in Eq. 46 is a useful statistic called the "variance," s^2, of the regression

$$s^2 = \frac{1}{n-j} \sum_{k=j+1}^{n} b_k^2 \tag{48}$$

The square root of this quantity, namely, the residual standard deviation, is an estimate of the experimental error.

Note that Eq. 46 and 48 contain components of B which have yet to be computed. It is known, however, that $\sum_{j=1}^{n} b_j^2 = \sum_{i=1}^{n} y_i^2$, from the Pythagorean theorem, and therefore the following relation may be used to compute the required components:

$$\sum_{k=j+1}^{n} b_k^2 = \sum_{k=1}^{n} y_k^2 - \sum_{k=1}^{j} b_k^2 \tag{49}$$

THE DESIGN OF EXPERIMENTS

In the preceding section, it was noted that useful inferences could be made about the components of the B vector in the special case of curvilinear regression. This cannot be demonstrated in the general case. To explain this, the pertinent orthogonal polynomials will be reviewed.

In the general case, Gram-Schmidt construction leads to the following relationships

$$\begin{aligned} p_{i1} &= d_{11} x_{i1} \\ p_{i2} &= d_{12} x_{i1} + d_{22} x_{i2} \\ p_{i3} &= d_{13} x_{i1} + d_{23} x_{i2} + d_{33} x_{i3} \\ &\quad \cdots \end{aligned} \tag{34}$$

and it is seen that each polynominal is a linear combination of the independent variables.

The difficulty lies in the fact that generally, there is, no physical significance in the particular linear combinations that arise. This is more obvious when one notes that these combinations arise strictly according to the manner in which the independent variable values were chosen and depend in no way on the functional relation between the dependent variable and the independent variables.

This statement may be verified by observing that the D matrix is formed independently of the Y matrix.

Fortunately, when the Gram-Schmidt construction is applied to the curvilinear regression model, each polynominal is one degree higher than the one preceding it

$$\begin{aligned} p_{i1} &= d_{11} \\ p_{i2} &= d_{12} + d_{22}\, x_i \\ p_{i3} &= d_{13} + d_{23}\, x_i + d_{33}\, x_i^{\,2} \\ &\;\;\ldots \end{aligned} \tag{37}$$

The nature of an nth degree polynomial is not altered by its having terms of degree less than n; therefore, useful inferences may be made.

It is so useful to form inferences, and to work in the undistorted P space, that the general case will be examined further. If only diagonal elements appeared in the D matrix, Eq. 34 could be rewritten

$$\begin{aligned} p_{i1} &= d_{11}\, x_{i1} \\ p_{i2} &= (0)\, x_{i1} + d_{22}\, x_{i2} \\ p_{i3} &= (0)\, x_{i1} + (0)\, x_{i2} + d_{33}\, x_{i3} \\ &\;\;\ldots \end{aligned}$$

In this speculative case, each p_{ij} is related to only one independent variable. Therefore, it would not be unreasonable to make inferences about the corresponding b_j components, namely, that b_j would show a functional relation between the dependent variable and the jth independent variable.

When would this speculative case arise? The off-diagonal terms in the D matrix appear during the Gram-Schmidt construction. The only case where they are zero is the one where the columns of the X matrix are already orthogonal.[15] This is unfortunate, because an orthogonal relationship among the independent variables is far from a general case.

However, the nature of an independent variable is that its values are free to be chosen (hence the term independent). By choosing these values so that the X matrix is orthogonal before the experiment is run, one gets a designed experiment. The D matrix can now be viewed as a description of the arbitrary way in which values are assigned to the independent variables.

For example, suppose that there are two brands of soap, brand P and brand Q. Suppose, further, that each manufacturer has recently introduced a new version of his brand containing the ingredient TRID. There are two independent variables: the brand, and the presence or absence of TRID. The value +1 can be assigned to the first independent variable (x_{i1}) when the soap is brand P, and the value -1 when the soap is brand Q. Furthermore, the value +1 can be assigned to the second independent variable (x_{i2}) when the product contains TRID and, -1 when it does not.

To determine the relative cleaning power of the four soaps, a test agency measures the dependent variable by measuring the percentage of light reflected from sheets washed in the various products. It is desirable to compute statistics, b_j, from which proper inference can be made about the two independent variables.

[15] But not necessarily normal

Note the X matrix formed when the following four data points are chosen

1. Brand P without TRID
2. Brand Q without TRID
3. Brand P with TRID
4. Brand Q with TRID

$$X = \begin{bmatrix} +1 & -1 \\ -1 & -1 \\ +1 & +1 \\ -1 & +1 \end{bmatrix} \qquad (50)$$

The two columns of X are already orthogonal, and P and D are

$$P = \begin{bmatrix} +\frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \\ +\frac{1}{2} & +\frac{1}{2} \\ -\frac{1}{2} & +\frac{1}{2} \end{bmatrix} \qquad D = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \qquad (51)$$

due entirely to normalizing. Hence

$$P_{.1} = \frac{1}{2}x_{.1}$$

$$P_{.2} = \frac{1}{2}x_{.2}$$

Components $b_1 = P_{.1} \cdot Y$ and $b_2 = P_{.2} \cdot Y$ may be inferred to be the variation of Y with brand name and with TRID, respectively. This is possible because a good choice of data points was made.

The $p_{.j}$ vectors in this case are called "orthogonal contrasts." This is only a simple example of the designed experiment. A specific class of designed experiments is discussed in the next section.

FACTORIAL ANALYSIS

One experimental design, leading to orthogonal relationships among the independent variables, is the lattice[16] design. The analysis of this design will be presented, and an example will be followed throughout the section.

Consider an experiment where there are two independent variables, x_1 and x_2. In this example, it is desirable that x_1 take three distinct values and that x_2 take four distinct values. A special situation, involving a total of 12 points, occurs when the four x_2 values are repeated for each of the three x_1 values. In this case, the paired values taken by the two independent variables may be arranged into the following lattice:

$$\begin{matrix} x_1(1),\ x_2(1) & x_1(1),\ x_2(2) & x_1(1),\ x_2(3) & x_1(1),\ x_2(4) \\ x_1(2),\ x_2(1) & x_1(2),\ x_2(2) & x_1(2),\ x_2(3) & x_1(2),\ x_2(4) \\ x_1(3),\ x_2(1) & x_1(3),\ x_2(2) & x_1(3),\ x_2(3) & x_1(3),\ x_2(4) \end{matrix} \qquad (52)$$

[16] A widely used special case of factorial analysis

where $x_1(1)$, $x_1(2)$, and $x_1(3)$ (which may be denoted $x_1(j_1)$, $1 \leq j_1 \leq 3$) are the three values taken by the first independent variable, and $x_2(j_2)$, $1 \leq j_2 \leq 4$, are the four values taken by the second independent variable. In such a design, j_1 is called the "level" of x_1, and j_2 is called the level of x_2. The variables x_1 and x_2 are called "factors."

In this experiment, the dependent variable y may be doubly subscripted by the level indices of the independent variables. The experimental data, therefore, may be arranged into the same lattice

$$\begin{array}{cccc} y_{11} & y_{.2} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \end{array} \tag{53}$$

To analyze such a design, a regression model must be set up. The first step in establishing a model is to "partition" the independent variables. Although this term is just now introduced in factorial analysis, it refers to substitutions already described informally in the sections entitled Orthogonal Polynomials and Curvilinear Regression, and the Design of Experiments. Formally, partitioning is the procedure whereby the expression $y = f(x)$ is replaced by the expression $y = f(\underline{x}^1(x), \underline{x}^2(x), \ldots, \underline{x}^n(x))$. In other words, the expression "y is a function of x" becomes "y is a function of a number of functions of x." This procedure introduces no restrictive assumptions.

A new notation is used in this formal definition for reasons that will become apparent as the model is developed. The symbol $\underline{x}^1$ refers to an arbitrary function of x. The superscript is an index which identifies the function; it is not an exponent. The form of these functions in the case of curvilinear regression is

$$\begin{aligned} \underline{x}^1 &= 1 \\ \underline{x}^2 &= x \\ \underline{x}^3 &= x^2 \end{aligned} \tag{54}$$

and so on, where, in the conventional notation on the right (not underscored), a superscript is indeed an exponent. This is a common example of partitioning, but certainly not the only example. The arbitrary functions may take such diverse forms as

$$\begin{aligned} \underline{x}^1 &= 1 \\ \underline{x}^2 &= \log(x) \\ \underline{x}^3 &= \tan(x) \end{aligned} \tag{55}$$

and so on. The basic point to remember is that for a finite number of x values, the arbitrary function $\underline{x}^i$ is only a set of functional values and may be viewed as a vector in linear vector space.

Sometimes only the set of functional values is given, and there is no analytic form to $\underline{x}^i$. An example of such a case may be taken directly from the section on the design of experiments:

$\underline{x}^1 = +1$ if soap is brand P $\qquad \underline{x}^2 = +1$ if TRID is contained

$\underline{x}^1 = -1$ if soap is brand Q $\qquad \underline{x}^2 = -1$ if TRID is not contained

The latter case is called a qualitative partition, and the former cases (such as Eq. 54 and 55) are called quantitative partitions.

The regression model for y, a function of one variable with partitioning, may now be written

$$y = a_1\underline{x}^1 + a_2\underline{x}^2 + \ldots + a_n\underline{x}^n$$

and a simple criterion for a linear model may be expressed

$$\frac{\partial y}{\partial \underline{x}^i} = a_i \text{ (a constant)} \qquad 1 \leq i \leq n \tag{56}$$

The linear approach to lattice designs requires that each independent variable (factor) be partitioned into as many arbitrary functions as there are functional values (levels). This is a controversial point. The nonlinear approach presented by many texts does not require partitioning (in fact, partitioning is not possible without reverting, at least part way, to a linear approach). An objection to the linear approach is that meaningful partitions cannot always be made. (Take three brands of cigarettes for an independent variable, as an example.) The objection is not valid, however, as meaningless partitions can always be performed and a "departitioning" performed at the end of the analysis. This will be discussed further after the development of the model has been completed and the analysis has been presented.

In the lattice design being followed, there are two factors, one with three levels and one with four levels:

$$x_1(j_1) \qquad 1 \leq j_1 \leq 3$$

$$x_2(j_2) \qquad 1 \leq j_2 \leq 4$$

The first factor must be partitioned into three functions

$${\underline{x}^1}^1(j_1),\ {\underline{x}^2}^1(j_1),\ {\underline{x}^3}^1(j_1) \qquad 1 \leq j_1 \leq 3$$

and the second must be partitioned into four

$${\underline{x}^1}^2(j_2),\ {\underline{x}^2}^2(j_2),\ {\underline{x}^3}^2(j_2),\ {\underline{x}^4}^2(j_2) \qquad 1 \leq j_2 \leq 4$$

to equal the number of levels.

The partitioning must be performed so that there is linear independence among the arbitrary functions and hence in their matrix of values. The matrix of values for the first factor is

$$\begin{bmatrix} {\underline{x}^1}^1(1) & {\underline{x}^2}^1(1) & {\underline{x}^3}^1(1) \\ {\underline{x}^1}^1(2) & {\underline{x}^2}^1(2) & {\underline{x}^3}^1(2) \\ {\underline{x}^1}^1(3) & {\underline{x}^2}^1(3) & {\underline{x}^3}^1(3) \end{bmatrix} \tag{57}$$

If $x_1(1) = 1$, $x_1(2) = 2$, and $x_1(3) = 3$, and this first factor is partitioned according to Eq. 54, the matrix of values to be checked for linear dependence is

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{bmatrix}$$

Once independent partitioning is accomplished for each factor, the regression model itself may be considered. For a lattice design of two factors, the general form of this model must be

$$y_{j_1 j_2} = f\left[x_1(j_1),\ x_2(j_2)\right]$$

When the partition device is applied, the model becomes

$$y_{j_1 j_2} = f\left[\underline{x}^{1\,1}(j_1), \underline{x}^{2\,1}(j_1), \underline{x}^{3\,1}(j_1), \underline{x}^{1\,2}(j_2), \underline{x}^{2\,2}(j_2), \underline{x}^{3\,2}(j_2), \underline{x}^{4\,2}(j_2)\right]$$

If the requirement is made that y be a linear function of x_1 when x_2 is constant, the following relationship may be derived from Eq. 56

$$\frac{\partial y}{\partial \underline{x}^{i_1 1}} = a_{i_1}(x_2) \text{ (a function only of } x_2\text{)} \quad 1 \leq i_1 \leq 3$$

Furthermore, if the requirement is made that y be a linear function of x_2 when x_1 is held constant, another relationship may be derived from Eq. 56

$$\frac{\partial y}{\partial \underline{x}^{i_2 2}} = a_{i_2}(x_1) \text{ (a function only of } x_1\text{)} \quad 1 \leq i_2 \leq 4$$

The most general solution to these differential relationships can be shown to be

$$\begin{aligned} y_{j_1 j_2} =\ & a_{11}\underline{x}^{1\,1}(j_1)\underline{x}^{1\,2}(j_2) + a_{12}\underline{x}^{1\,1}(j_1)\underline{x}^{2\,2}(j_2) + a_{13}\underline{x}^{1\,1}(j_1)\underline{x}^{3\,2}(j_2) + a_{14}\underline{x}^{1\,1}(j_1)\underline{x}^{4\,2}(j_2) \\ & + a_{21}\underline{x}^{2\,1}(j_1)\underline{x}^{1\,2}(j_2) + a_{22}\underline{x}^{2\,1}(j_1)\underline{x}^{2\,2}(j_2) + a_{23}\underline{x}^{2\,1}(j_1)\underline{x}^{3\,2}(j_2) + a_{24}\underline{x}^{2\,1}(j_1)\underline{x}^{4\,2}(j_2) \quad (58) \\ & + a_{31}\underline{x}^{3\,1}(j_1)\underline{x}^{1\,2}(j_2) + a_{32}\underline{x}^{3\,1}(j_1)\underline{x}^{2\,2}(j_2) + a_{33}\underline{x}^{3\,1}(j_1)\underline{x}^{3\,2}(j_2) + a_{34}\underline{x}^{3\,1}(j_1)\underline{x}^{4\,2}(j_2) \end{aligned}$$

provided that one arbitrary function of each independent variable is merely a constant. The practice of setting $\underline{x}^{1\,1}$ and $\underline{x}^{1\,2}$ (and for that matter the first arbitrary function of any factor) to a constant value of unity is so common and so necessary for this solution that it will be assumed throughout the rest of the text

This is, finally, the regression model for the 3 × 4 lattice design being used as an example. That it obeys the differential relations above may be verified by substitution.

As a further demonstration of the linearity of the model, Eq. 58 may be expanded into 12 linear equations, in 12 unknowns, by substituting in the 12 lattice points. [The data ($y_{j_1 j_2}$ from Eq. 53) are directly substituted, but the values of the independent variables ($x_1(j_1)$, $x_2(j_2)$ from Eq. 52) must, of course, be transformed according to the partitioning used (such as in Eq. 54).[17]] The $a_{i_1 i_2}$ may, therefore, be determined by the techniques of the previous sections However, a more powerful technique is possible and will be described later.

There are some interesting and desirable properties of this model (Eq. 58) which should be noted. The terms in the equation may be arranged into the same lattice as the design points (Eq. 53). This property is a key to the analysis which will be presented. Another property of

[17] See Eq. 59 for a symbolic example of this substitution.

the model, and one which makes it realistic, is that when each factor is partitioned into zero-order, linear, quadratic, etc., functions (Eq. 54), the model takes the form of a Taylor's expansion of y in two variables

$$
\begin{aligned}
y(x_1, x_2) = \\
& a_{11} + a_{12} x_2 + a_{13} x_2^2 + a_{14} x_2^3 \\
& + a_{21} x_1 + a_{22} x_1 x_2 + a_{23} x_1 x_2^2 + a_{24} x_1 x_2^3 \\
& + a_{31} x_1^2 + a_{32} x_1^2 x_2 + a_{33} x_1^2 x_2^2 + a_{34} x_1^2 x_2^3
\end{aligned}
\tag{59}
$$

Here the level indices have been stripped to enhance the form of the expansion, and $a_{i_1 i_2}$ is an approximation for

$$\left(\frac{\partial}{\partial x_1}\right)^{i_1-1} \left(\frac{\partial}{\partial x_2}\right)^{i_2-1} \frac{y\,(0,0)}{(i_1-1)!\,(i_2-1)!}$$

Because of the finite number of points in the lattice, only the low order terms of the expansion appear.

The model given may be simplified (at least in appearance) as follows:

$$\underline{Y}_{j_1 j_2} = \underline{X}^{i_1}_{j_1}\ \underline{X}^{i_2}_{j_2}\ \underline{A}_{i_1 i_2} \tag{60}$$

where the underscored capital letters are tensors.

Tensor notation (Ref. 6) is intoduced at this point because it is particularly suited to the description of the analysis. The small amount of tensor notation used will be fully explained. The four variables in Eq. 60 are two-dimensional arrays similar to matrices, except that some of the indices appear as superscripts. The elements of the $\underline{Y}_{j_1 j_2}$ array appear in Eq. 53.

The elements of $\underline{X}^{i_1}_{j_1}$ are $\underline{x}^{i_1}(j_1)$, where[18]

$$\begin{cases} 1 \leq i_1 \leq 3 \\ 1 \leq j_1 \leq 3 \end{cases}$$

and those of $\underline{X}^{i_2}_{j_2}$ are $\underline{x}^{i_2}(j_2)$, where

$$\begin{cases} 1 \leq i_2 \leq 4 \\ 1 \leq j_2 \leq 4 \end{cases}$$

The elements of the $\underline{A}_{i_1 i_2}$ tensor span the a coefficient of Eq. 58.

Tensors may be multiplied in two different ways to obtain either an inner or an outer product. Only the inner product is of interest in this development, and only that product will appear.

As a first example of the tensor product, consider the product $\underline{X}^{i_1}_{j_1}\ \underline{X}^{i_2}_{j_2}$ from Eq. 60. This product yields a four-dimensional array which may be denoted $\underline{X}^{i_1 i_2}_{j_1 j_2}$ without notational ambiguity.

[18] These elements may be recognized as the matrix of values (Eq. 57).

The $i_1 i_2 j_1 j_2$th element of this array is the product of the $i_1 j_1$th element from $\underline{X}^{i_1}_{j_1}$ and the $i_2 j_2$th element from $\underline{X}^{i_1}_{j_2}$.

The continuation of the product in Eq. 60 presents a second example of the tensor product

$$\underline{X}^{i_1 i_2}_{j_1 j_2} \underline{A}_{i_1 i_2}$$

Note that the superscripts of $\underline{X}$ appear as the subscripts of $\underline{A}$.

There are two steps in forming this product. The first is forming the following four-dimensional array (there are six indices, but only four are distinct):

$$\underline{Y}^{i_1 i_2}_{j_1 j_2 i_1 i_2}$$

where the $i_1 j_1 i_2 j_2$th element of this array is formed from the product of the $i_1 j_1 i_2 j_2$th element of $\underline{X}$ and the $i_1 i_2$th element of $\underline{A}$.

The inner product is completed by a second step where the indices appearing both as subscripts and as superscripts are to be summed out as follows:

$$\underline{Y}_{j_1 j_2} = \sum_{i_1} \sum_{i_2} \underline{Y}^{i_1 i_2}_{j_1 j_2 i_1 i_2} \tag{61}$$

which reduces the product to a two-dimensional array. A numerical example of such a tensor product appears in Appendix E.

Returning to the subject of factorial analysis, in Eq. 60 there appears a tensor for each independent variable or factor, $\underline{X}^{i_1}_{j_1}$ and $\underline{X}^{i_2}_{j_2}$. While their product can be immediately written as $\underline{X}^{i_1 i_2}_{j_1 j_2}$, it is better to analyze them separately. Each $\underline{X}$ tensor can be analyzed in a manner similar to the matrix methods given previously.

It is desirable to separately transform each of the $\underline{X}$ tensors into orthonormal $\underline{P}$ tensors. This may be done by the Gram-Schmidt construction, which was previously described by the matrix transformation

$$P = X\,D$$

This transformation may be rewritten in tensor notation

$$\underline{P}^s_j = \underline{X}^i_j \, \underline{D}^s_i \tag{62}$$

Some insight into tensors can be gained by studying this equation. Because it describes the same transformation as the matrix equation, it can be observed that the superscript of the tensor embodies the row property of the matrix[19] and the subscript of the tensor embodies the column property.[20]

Consider now a tensor $\underline{C}^k_s$, such that the tensor product

$$\underline{D}^s_i \, \underline{C}^k_s = \underline{I}^k_i$$

where the elements of $\underline{I}$ are unity if $i = k$ and are zero if $i \neq k$. Such a tensor $\underline{C}$ contains the inverse elements of the $\underline{D}$ array in the matrix sense.

[19] Also called the contravariant property.

[20] Also called the covariant property.

The tensor $\underline{I}$ has the property that, when it multiplies another tensor such as $\underline{X}$, the following relation results:

$$\underline{X}_j^i \underline{I}_i^k = \underline{X}_j^k$$

Performing this multiplication according to the inner product rules will show that the individual elements of $\underline{X}_j^k$ are equal to the individual elements of $\underline{X}_j^i$, so the change is only symbolic. This relationship will be used below.

Both sides of Eq. 62 may be multiplied by $\underline{C}$ to give the following:

$$\underline{P}_j^s \underline{C}_s^k = \underline{X}_j^i \underline{D}_i^s \underline{C}_s^k = \underline{X}_j^k \tag{63}$$

A further multiplication by $\underline{I}_k^i$ gives

$$\underline{P}_j^s \underline{C}_s^i = \underline{X}_j^i \tag{64}$$

This relation, with the indices properly subscripted, may be substituted into Eq. 60 for each factor $\underline{X}$ to give

$$\underline{Y}_{j_1 j_2} = \underline{P}_{j_1}^{s_1} \underline{C}_{s_1}^{i_1} \underline{P}_{j_2}^{s_2} \underline{C}_{s_2}^{i_2} \underline{A}_{i_1 i_2} \tag{65}$$

Some desirable properties of tensors allow a rapid completion of the analysis. Tensors, unlike matrices, obey the commutative rule as well as the associative rule of multiplication; for example

$$\underline{P}_j^s \underline{C}_s^i = \underline{C}_s^i \underline{P}_j^s \tag{66}$$

Therefore, one may rearrange the product terms in Eq. 65 at will. One way follows:

$$\underline{Y}_{j_1 j_2} = \underline{P}_{j_1}^{s_1} \underline{P}_{j_2}^{s_2} \underline{B}_{s_1 s_2} \tag{67}$$

where

$$\underline{B}_{s_1 s_2} = \underline{C}_{s_1}^{i_1} \underline{C}_{s_2}^{i_2} \underline{A}_{i_1 i_2}$$

It is epeated at this point that each $\underline{P}$ tensor is a two-dimensional array which contains the same elements as an orthonormal matrix P. A property of such matrices is that $P'P = I$. This relation may also be rewritten in tensor notation

$$\underline{P}_t^j \underline{P}_j^s = \underline{I}_t^s$$

It is legitimate here to use the same symbol for each $\underline{P}$ tensor, because each contains the same elements. However, the index which describes the column property[21] in the original $\underline{P}_j^s$ array must describe the row property in the new $\underline{P}_t^j$ array.

Hence j is a subscript in the tensor corresponding to the matrix P, and is a superscript in the tensor corresponding to the matrix P.′ It is in this way that tensors are transposed. Both sides

[21] This is because the matrix P was defined so that it was column-orthogonal.

of Eq. 67 may be multiplied by $\underline{P}^{j_1}_{s_1}$ and $\underline{P}^{j_2}_{s_2}$ to give the following:[22]

$$\underline{P}^{j_1}_{t_1}\,\underline{P}^{j_2}_{t_2}\,\underline{Y}_{j_1 j_2} = \underline{B}_{t_1 t_2} \tag{68}$$

The tensor $\underline{A}$ may be solved for in a manner analogous to the matrix solution of Eq. 25. However, as before, this will be considered an anticlimax in the analysis, and, in fact, it is very seldom done in factorial analyis.

Equation 68 may be compressed by the rule of tensor products as follows:

$$\underline{P}^{j_1 j_2}_{t_1 t_2}\,\underline{Y}_{j_1 j_2} = \underline{B}_{t_1 t_2} \tag{69}$$

The tensor $\underline{P}$ operates on the data tensor $\underline{Y}$, to yield a third tensor $\underline{B}$. A transformation is resented in the next section which changes $\underline{Y}$ and $\underline{B}$ into one-dimensional arrays while preserving the total number of elements contained. The same transformation, when applied to the $\underline{P}$ tensor, reduces it to a two-dimensional array. It will be shown, further, that this latter array s also orthonormal. This will establish that the elements of the $\underline{B}$ tensor are components of the lata $\underline{Y}$. Therefore, the theorem of Pythagoras applies, which, for the example, is expressed

$$\sum_{j_1}\sum_{j_2}\left(\underline{y}_{j_1 j_2}\right)^2 = \sum_{t_1}\sum_{t_2}\left(\underline{b}_{t_1 t_2}\right)^2 \tag{70}$$

vhere a double summation is required because of the double index.

Furthermore, the elements of $\underline{B}$ may be inferred to be variations of the data due to certain ariations in the independent variables. To complete the analysis. it is necessary to identify hese variations. The key to this identification is the double subscript of the $\underline{B}$ tensor. Just s the double subscript of the $\underline{Y}$ tensor identifies the individual data with the levels taken by he independent variables, the subscripts of the $\underline{B}$ tensor (which run the same gamut as the Y ubscripts) identify the components with these variations in the independent variables.

A study of this can be made by expanding Eq. 67 according to the definition of the tensor roduct

$$\begin{aligned}
\underline{y}_{j_1 j_2} =\ & \\
& \underline{b}_{11}\underline{p}^{1\,1}(j_1)\,\underline{p}^{1\,2}(j_2) + b_{12}\underline{p}^{1\,1}(j_1)\,\underline{p}^{2\,2}(j_2) + \underline{b}_{13}\underline{p}^{1\,1}(j_1)\,\underline{p}^{3\,2}(j_2) + \underline{b}_{14}\underline{p}^{1\,1}(j_1)\,\underline{p}^{4\,2}(j_2) + \\
& \underline{b}_{21}\underline{p}^{2\,1}(j_1)\,\underline{p}^{1\,2}(j_2) + \underline{b}_{22}\underline{p}^{2\,1}(j_1)\,\underline{p}^{2\,2}(j_2) + \underline{b}_{23}\underline{p}^{2\,1}(j_1)\,\underline{p}^{3\,2}(j_2) + \underline{b}_{24}\underline{p}^{2\,1}(j_1)\,\underline{p}^{4\,2}(j_2) + \qquad (71) \\
& \underline{b}_{31}\underline{p}^{3\,1}(j_1)\,\underline{p}^{1\,2}(j_2) + \underline{b}_{32}\underline{p}^{3\,1}(j_1)\,\underline{p}^{2\,2}(j_2) + \underline{b}_{33}\underline{p}^{3\,1}(j_1)\,\underline{p}^{3\,2}(j_2) + \underline{b}_{34}\underline{p}^{3\,1}(j_1)\,\underline{p}^{4\,2}(j_2)
\end{aligned}$$

[22] The difference between $\underline{B}_{s_1 s_2}$ and $\underline{B}_{t_1 t_2}$ is only symbolic; they each contain the same array of umbers.

As an example, let the independent variables be partitioned according to Eq. 54.

$$\underline{x}^{1\,1}(j_1) = 1 \qquad \underline{x}^{1\,2}(j_2) = 1$$

$$\underline{x}^{2\,1}(j_1) = x_1 \qquad \underline{x}^{2\,2}(j_2) = x_2$$

$$\underline{x}^{3\,1}(j_1) = x_1{}^2 \qquad \underline{x}^{3\,2}(j_2) = x_2{}^2$$

$$\underline{x}^{4\,2}(j_2) = x_2{}^3$$

Then from Eq. 62

$$\underline{p}^{1\,1}(j_1) = \underline{d}^{1\,1}_{1_1} \qquad \underline{p}^{1\,2}(j_2) = \underline{d}^{1\,2}_{1_2}$$

$$\underline{p}^{2\,1}(j_1) = \underline{d}^{1\,1}_{2_1} + \underline{d}^{2\,1}_{2_1}\underline{x}_1 \qquad \underline{p}^{2\,2}(j_2) = \underline{d}^{1\,2}_{2_2} + \underline{d}^{2\,2}_{2_2}\underline{x}_2$$

$$\underline{p}^{3\,1}(j_1) = \underline{d}^{1\,1}_{3_1} + \underline{d}^{2\,1}_{3_1}\underline{x}_1 + \underline{d}^{3\,1}_{3_1}\underline{x}_1^2 \qquad \underline{p}^{3\,2}(j_2) = \underline{d}^{1\,2}_{3_2} + \underline{d}^{2\,2}_{3_2}\underline{x}_2 + \underline{d}^{3\,2}_{3_2}\underline{x}_2^2$$

$$\underline{p}^{4\,2}(j_2) = \underline{d}^{1\,2}_{4_2} + \underline{d}^{2\,2}_{4_2}\underline{x}_2 + \underline{d}^{3\,2}_{4_2}\underline{x}_2^2 + \underline{d}^{4\,2}_{4_2}\underline{x}_2^3$$

Therefore, $\underline{p}^{1\,1}$ and $\underline{p}^{1\,2}$ are zero-order functions; $\underline{p}^{2\,1}$ and $\underline{p}^{2\,2}$ are linear functions of their respective factors; $\underline{p}^{3\,1}$ and $\underline{p}^{3\,2}$ are quadratic functions of their respective factors; and $\underline{p}^{4\,2}$ is a cubic function of x_2.

An inspection of Eq. 71 will show that an element $\underline{b}_{s_1 s_2}$ of the $\underline{B}$ tensor shows the variation in the data $\underline{Y}$, due to the variation in $\underline{p}^{s\,1}$ and in $\underline{p}^{s\,2}$. A component which depends on two variations is called an interaction component. In a completely general case, all $\underline{b}_{s_1 s_2}$ are interaction components. However, in practical cases, such as the example being given, the independent variables are partitioned in such a way that the first orthogonal function, p^1 of each is zero order, or constant. Because of this, certain degenerations occur. The component $\underline{b}_{11}$ is inferred to be the variation of the data due to the zero-order variation of x_1 and of x_2. This interaction is doubly degenerate and is itself zero order. The term $\underline{b}_{12}$ shows the variation of the data resulting from the zero-order variation of x_1 and the linear variation of x_2. This interaction is singly degenerate and is called a main effect, namely, the linear x_2 effect. The rest of the inferences are found in the same manner and are tabulated in Table 1.

Henceforth, only interactions which do not degenerate will be termed interactions.

An inference need not always be made about a component. In order to test whether components are showing a true functional relation between y and the independent variables, it is necessary to have extra components where no functional relation is expected to appear. These components are of the same nature as those described in the section on curvilinear regression and are termed error components. An F test can be made where each component, about which an inference is made, is tested against the error components.

In the example given, there are not enough components to test all possible inferences. Two approaches are possible. First, it may be that there are no true functional relations expected

TABLE 1. Inferences for a 3 x 4 Experiment With Linear, Quadratic, and Cubic Partitioning

Component	x_1 Variation	x_2 Variation	Inferences
B_{11}	zero order	zero order	zero order[a]
B_{12}	zero order	linear	lin. x_2 effect[b]
B_{13}	zero order	quadratic	quad. x_2 effect[b]
B_{14}	zero order	cubic	cub. x_2 effect[b]
B_{21}	linear	zero order	lin. x_1 effect[b]
B_{22}	linear	linear	lin. x_1 lin. x_2 interaction
B_{23}	linear	quadratic	lin. x_1 quad. x_2 interaction
B_{24}	linear	cubic	lin. x_1 cub. x_2 interaction
B_{31}	quadratic	zero order	quad. x_1 effect[b]
B_{32}	quadratic	linear	quad. x_1 lin. x_2 interaction
B_{33}	quadratic	quadratic	quad. x_1 quad. x_2 interaction
B_{34}	quadratic	cubic	quad. x_1 cub. x_2 interaction

[a] Doubly degenerate.
[b] Singly degenerate.

among the interaction components. In this case, the F ratio used to test a true functional relation between y and a linear variation in x_1 is computed as follows:[23]

$$F(1,6) = \frac{(\underline{b}_{21})^2}{\frac{1}{6}\left[(\underline{b}_{22})^2 + (\underline{b}_{23})^2 + (\underline{b}_{24})^2 + (\underline{b}_{32})^2 + (\underline{b}_{33})^2 + (\underline{b}_{34})^2\right]} \qquad (72)$$

and for a linear relation between y and x_2

$$F(1,6) = \frac{(\underline{b}_{12})^2}{\frac{1}{6}\left[(\underline{b}_{22})^2 + (\underline{b}_{23})^2 + (\underline{b}_{24})^2 + (\underline{b}_{32})^2 + (\underline{b}_{33})^2 + (\underline{b}_{34})^2\right]} \qquad (73)$$

and so on for quadratic and cubic relations. If it is necessary to infer that interaction effects do exist, then it is necessary to create more components, which is the second approach. The only way to do this is to add a new factor. This factor may be some new variable in the analysis, or it may be a factor about which no inference is made. The latter case is called multiple replication, because it is merely repetition of the experiment.

It is not always possible to partition a factor in a meaningful manner. A "brand name" factor was given as an example. If factor x_1 of that example had re)resented brands A, B, and C of automobile tires, then it would be meaningless to talk about the linear and quadratic variation among the brands. However, one can still make the meaningless partition and get the total variation of y, due to brands by adding (vectorially) the two meaningless components

$$\vec{\underline{b}}_{21} + \vec{\underline{b}}_{31} = \vec{\underline{b}}_{\text{brands, 1}}$$

Because $\underline{b}_{21}$ and $\underline{b}_{31}$ are components in orthogonal directions, the length of the new vector is

$$\sqrt{(\underline{b}_{21})^2 + (\underline{b}_{31})^2}$$

by the Pythagorean theorem. This may be called departitioning.

[23] Note that the denominator contains the interaction components shown in Table 1.

If one assumes, as before, that the interaction terms are error, the F ratio to test the variation of y due to brands is

$$F(2,6) = \frac{\frac{1}{2}\left[(\underline{b}_{21})^2 + (\underline{b}_{31})^2\right]}{\frac{1}{6}\left[(\underline{b}_{22})^2 + (\underline{b}_{23})^2 + (\underline{b}_{24})^2 + (\underline{b}_{32})^2 + (\underline{b}_{33})^2 + (\underline{b}_{34})^2\right]} \tag{74}$$

where the general rule is applied that the F function is the ratio of the mean square components being tested to the mean square components considered to be due to experimental error.

The choosing of the components to be tested, and of the F ratios to be formed, is called the summary of the analysis and, as stated before, is strictly within the working area of the statistician.

Thus far, only a particular two-factor experiment has been considered. Generalizing to the n factor experiment is remarkably simple in tensor notation and is summarized below.

SUMMARY OF THE ANALYSIS OF LATTICE DESIGNS

Assume the regression model

$$\underline{Y}_{j_1 j_2 j_3 \cdots} = \underline{X}^{i_1}_{j_1} \underline{X}^{i_2}_{j_2} \underline{X}^{i_3}_{j_3} \cdots \underline{A}_{i_1 i_2 i_3 \cdots} \tag{75}$$

1. Partition each independent variable to obtain the matrix of values which are to be inserted into the individual $\underline{X}$ tensors.

$\underline{x}^{i_1}(j_1) = x^{i_1}[x_1(j_1)]$ (each $\underline{x}^{i_1}$ an arbitrary function of x_1)

$\underline{x}^{i_2}(j_2) = x^{i_2}[x_2(j_2)]$ (each $\underline{x}^{i_2}$ an arbitrary function of x_2)

$\underline{x}^{i_3}(j_3) = x^{i_3}[x_3(j_3)]$ (each $\underline{x}^{i_3}$ an arbitrary function of x_3)

. . .

where, for each factor, the number of values taken by i (number of functions) is equal to the number of values taken by j (number of levels) to obtain a square matrix of values.

2. If each matrix of values possesses linear independence, form the orthonormal tensors by the Gram-Schmidt construction

$$\underline{P}^{s_1}_{j_1} = \underline{X}^{i_1}_{j_1} \underline{D}^{s_1}_{i_1}$$

$$\underline{P}^{s_2}_{j_2} = \underline{X}^{i_2}_{j_2} \underline{D}^{s_2}_{i_2}$$

$$\underline{P}^{s_3}_{j_3} = \underline{X}^{i_3}_{j_3} \underline{D}^{s_3}_{i_3}$$

. . .

3. Transpose each $\underline{P}$ and, using the solution process of Eq. 67 and Eq. 68, evaluate the following equation

$$\underline{B}_{s_1 s_2 s_3 \cdots} = \underline{P}^{j_1 j_2 j_3 \cdots}_{s_1 s_2 s_3 \cdots} \underline{Y}_{j_1 j_2 j_3 \cdots} \tag{76}$$

4. Form inferences about $\underline{B}$ (which may now contain three-way and higher interactions) and summarize according to the desires of the statistician.

The factorial analysis program in Appendix C uses this method. Other facets of this program will appear in succeeding sections.

The reader is referred to Ref. 7 for another description of the linear approach to lattice designs.

TRANSFORMATION OF INDICES

As stated in the preceding section, a transformation will be described which reduces the multidimensional $\underline{Y}$ and $\underline{B}$ arrays to one-dimensional arrays. The transformation works on the indices of the tensor, but it does not alter the number of elements in the array or their numerical values.

This transformation is of more than academic interest, because the addressable memory of a digital computer is one-dimensional. So, an n-dimensional array must be represented by a one-dimensional array, the elements of which are found by a single subscript computed from the many subscripts of the n-dimensional array.

Assume, for example, that there are m factors with the number of levels denoted by f_1, f_2, . . . , f_m. The total number of data points, n, is equal to the product of the fs. Consider the data tensor $\underline{Y}_{j_1 j_2 \cdot \cdot j_m}$, a variable with m subscripts. A single subscript j may be computed from these subscripts as follows:[24]

$$j-1 = j_1-1 + (j_2-1)f_1 + (j_3-1)f_1 f_2 + (j_4-1)f_1 f_2 f_3 + \dots + (j_m-1)f_1 f_2 \dots f_{m-1} \qquad (77)$$

The above formula[25] will be recognized immediately by one who has coded operations on m-dimensional arrays in digital computing machine language, and the number theorist will recognize it as a representation of $j-1$ in a hybrid-base number system. That is, a number may be written $(j_m-1) \dots (j_2-1)\,(j_1-1)$, which does not represent $j-1$ in a decimal or a binary number system, but in a system where the low order digit, (j_1-1), is of base f_1; the second order digit, (j_2-1), is of base f_2; and so on. Because the last viewpoint delivers the greatest insight into the transformation, an example will be given.

Consider a three-factor experiment where factor x_1 has two levels, x_2 has three levels, and x_3 has four levels. Table 2 illustrates how the three-dimensional array may be "stretched out" into a single array. An important property of the transformation in Table 2 is that it is a one-to-one transformation. That is, for every j there is a unique set, j_1, j_2, . . . , j_m, and vice versa. The uniqueness may be proved by induction in a manner similar to the proof in Ref. 8.

[24] This j is notationally distinct because it is not subscripted.

[25] A cumbersome translation, $j \rightarrow j-1$, is used because the FORTRAN language does not allow indices to run from zero.

TABLE 2. Transformation of Indices for a 2 x 3 x 4 Experiment Showing the Base—10 and Hybrid—Base Representation of the Single Index Formed

J_1-1	J_2-1	J_3-1	J-1 (Base 10)	J-1 (Hybrid base)
0	0	0	0	000
1	0	0	1	001
0	1	0	2	010
1	1	0	3	011
0	2	0	4	020
1	2	0	5	021
0	0	1	6	100
1	0	1	7	101
0	1	1	8	110
1	1	1	9	111
0	2	1	10	120
1	2	1	11	121
0	0	2	12	200
1	0	2	13	201
0	1	2	14	210
1	1	2	15	211
0	2	2	16	220
1	2	2	17	221
0	0	3	18	300
1	0	3	19	301
0	1	3	20	310
1	1	3	21	311
0	2	3	22	320
1	2	3	23	321

This transformation may be applied to the subscripts j and t in Eq. 69 as follows:

$$\underline{P}_t^j \underline{Y}_j = \underline{B}_t \tag{78}$$

It is important to recognize that Eq. 78 describes the same arithmetic computations as Eq. 69. In fact, the computer programmer will recognize Eq. 78 as the method of coding the computations of Eq. 69 in machine language.

Equation 78 may be immediately rewritten in matrix notation

$$P'Y = B \tag{79}$$

noting that the simplified tensor products of Eq. 78 corresponds exactly to a matrix product. Equation 67 may be transformed in the same manner

$$Y = P\,B \tag{80}$$

applying first the rule of tensor products to obtain $\underline{Y}_{j_1 j_2} = \underline{P}_{j_1 j_2}^{s_1 s_2} \underline{B}_{s_1 s_2}$.

The only way in which Eq. 79 and 80 may both be true is that

$$P' P = I \tag{81}$$

Thus it is shown that the matrix P. formed by the transformation of subscripts and superscripts of the tensor $\underline{P}_{j_1 j_2}^{s_1 s_2}$, is orthonormal It is therefore possible to observe two things. First, the lattice design fulfills the desired requirement that the independent variables be mutually independent (orthogonal). Second, the $\underline{b}_{s_1 s_2 \cdots}$ are truly components of the data $\underline{y}_{j_1 j_2 \cdots}$, and the inferences of the previous section are correct.

FRACTIONAL REPLICATION

The lattice designs already described are not the only means by which an orthogonal relationship may be established among the independent variables of an experiment. It is not within the scope of this report to describe the others in great detail. One type, however, is so common, and so intimately related to lattice designs, that something should be said about what it is, why it is used, and how it may be handled on digital computers.

Fractional replication is related to what is called the redundancy check. The reader will probably be familiar with the parity check on binary numbers or with check sums on punched cards. The redundancy check may be described as taking essential information and including with it redundant information computed from the essential information in some predetermined way. The purpose of this, of course, is to detect transmission errors (or, when the human element is involved, to reduce the possibility of forgery of numerical information). One may, for instance, look at the 10-digit serial number on his oil company card and note that the company could not possibly have 10 billion credit customers as this number implies. If some of these numbers are used as a redundancy check, one's chances of forging a legitimate serial number are small. A redundancy check may be added to the subscripts of data tensors from lattice designs. A particular class of redundancy checks will be described which requires that two or more factors have the same number of levels. For example, consider a simple two-factor experiment where each factor has three levels. The $\underline{Y}$ tensor than has nine elements

$$y_{11}, y_{12}, y_{13}, y_{21}, y_{22}, y_{23}, y_{31}, y_{32}, y_{33}$$

A third subscript may be added which has the property that the sum of the subscripts modulo 3 is zero

$$y_{111}, y_{123}, y_{132}, y_{213}, y_{222}, y_{231}, y_{312}, y_{321}, y_{333}$$

Consider now the case where a new three-level variable is introduced which takes the level suggested by the third subscript. In this case, some interesting things happen. First, the third subscript is now a redundancy check in mathematical form only. Actually, it achieves the same status as the other two subscripts. Second, a three-factor experiment has resulted, but something is missing, namely, those elements of a 27-element data tensor, the sums of whose subscripts equal 1 and 2 modulo 3. In fact, there is one-third of a 3 × 3 × 3 experiment (written $1/3 \times 3^3$). This is called a fractional replicate of the full 3^3 experiment.

The three subscripts possess the property of symmetry. That is, if originally there were only the second and third subscripts, the value of the first would be determined bv the same rule. In fact, if the first subscript is dropped, the second and third take the same set of values that the first and second did originally (but not in the same order); that is, all the points on a 3 × 3 lattice appear. Similarly, the second subscript is a redundancy check on the first and third, and if this second subscript is dropped, the first and third contain all the points on a 3 × 3 lattice (and again, the order is different).

Because it is a lattice design, the original two factors have an orthogonal relationship between them. But, by the property of symmetry, it must be concluded that the third is orthogonal to each of the original two. Thus, the fractional replicate is a designed experiment.

However, a paradox exists. Although there are three subscripts, there is only a two-dimensional array. In fact, the analysis made for complete factorial designs may be made here, ignoring any one of the three subscripts (because, as already noted, in all three cases the remaining subscripts describe a complete lattice). But each process analyzes only two factors. What happens to the third?

The solution tensor *B* consists of nine elements: the zero-order components of the data, the two main effects of the first factor, the two main effects of the second factor, and four interaction components. Because the vector basis is complete, the components related to the third factor must be contained somewhere. By its very nature, the zero-order component is eliminated, and, as already noted, the three main factors are mutually orthogonal. Therefore, the two components of the third factor must lie along some of the interaction components of the other two factors.

In this case, each factor is said to be "confounded" with the interaction of the other two factors This dual identity of components presents a problem. How does one know whether a component results from one factor or the interaction of two others? Proper inference can solve the problem. One may have good reason to believe that there are really no interaction effects. He then infers the components to be only main effects

One way to have confidence in such inferences is to confound factors with high-order interactions. (In the previous section, the regression model was described as an estimate of Taylor's expansion of the dependent variable. One therefore expects components resulting from high-order effects, even when real, to be small, because of the reciprocal factorials in the high-order terms of the expansion.) The only way to confound high-order interactions, in the class of fractional replication being described, is to have many factors with the same number of levels. Consider, for instance, a 4 × 4 × 4 × 4 experiment. The independent variable will have four subscripts, each taking values 1 to 4. One may add a fifth subscript with a redundancy check, so that the sum of the five must be 0 modulo 4, and may thus introduce a fifth factor. Each factor will be confounded with a four-way interaction. One then infers confidently that the components show only the functional relation of the factor and not a four-way interaction.

This example also illustrates the principal advantage of fractional replication. With four factors, one already had a need for 256 experimental points. Without fractionating, a fifth factor would require 1024; but with it, the need is still for only 256.

The type of fractional replication described above is called the Latin-square class, because the simplest cases lead to Latin-square patterns. For instance, from the $1/3 \times 3^3$ example, one may set up the following square:

	1	2	3
1	1	3	2
2	3	2	1
3	2	1	3

where the first subscript appears across the top, the second along the side, and the third in the boxes.

There are, of course, more complex examples in the Latin-square class; for instance, a $1/3 \times 4 \times 3^3$ experiment. The four-level factor has no redundancy check and hence no confounding with other factors. The other three, considered separately, form the classical Latin square above.

There are other types of redundancy checks which give rise to other factorial models. In some cases, there can be two independent redundancy checks on a number of factors of the same number of levels. This is called the Greco-Latin-square class of fractional replication. Other types of checks exist which give rise to incomplete-block and other exotic designs, some of which can be solved by the programming in appendixes A, B, and C and all of which can be solved by the methods of this report.

There are two ways of analyzing fractional replicates by the methods described under Factorial Analysis. One may be called "putting the burden on the statistician" and the other "putting the burden on the computing machine." In the first method, the redundant subscripts are ignored, and the regular factorial analysis is made, which computes each single degree of freedom. The statistician, knowing the redundancy check, then searches among the high-order components for the missing factors. The second method requires that the rest of the lattice, of which there is only a fraction, be filled in with zeros. The regular factorial analysis is then made, and highest-order interactions are successively discarded until there are fewer components than data. The remaining components are divided by the fraction of the experiment. In this method, all components are properly identified as described under Factorial Analysis.

As an example of the second method, consider the $1/3 \times 3^3$ experiment. This is first converted into a 3^3 by filling those elements of a $3 \times 3 \times 3$ lattice that are not in the design, with zeros. The factorial analysis is then made. The two- and three-way interaction components are discarded, and the main effects are divided by 1/3. Care must be taken in calculating F ratios in this case. The denominator must be calculated in the manner of Eq. 46, where the squared components to be tested, including the zero-order component, are subtracted from the total sum of squares of the data.

THE MISSING-VALUE PROBLEM

It sometimes happens that one or more points of a desired experiment are missing. This can happen for any number of reasons, such as a failure in recording instruments or an operator error. What is worse, in some cases these points are irreplaceable.

When there are points missing, there are two possible approaches to analyzing the data. One approach is to compute estimates of the missing value on the basis of the components of the data about which real functional relationships are inferred. The other approach is to revert to the general regression model.

There are disadvantages in both approaches, and both are subject to much criticism. In the first method, there is the disadvantage that the solution to the regression model that is formed after the computation of missing values is no longer the best solution in the sense of least squares. It has the advantage, however, that the inferences are preserved.

In reverting to the general regression model, one has the advantage of an exact solution in the sense of least squares, but the inferences become distorted and F tests are not properly made. Another advantage here is that it is a noniterative approach, whereas the missing-value approach remires iteration if there are two or more missing values. On a digital computer, this last is not an important consideration

The method of Villars (Ref. 9) has served the authors well as a missing-value technique and has been coded in machine language for the IBM 704 (Ref. 10). Unfortunately, it has not been updated to FORTRAN.

FINAL REMARKS

FORTRAN-coded programs for general regression, curvilinear regression, and factorial analysis, using the methods of the text, appear in Appendixes A, B, and C. Although these programs have output summaries, they cannot be described as final. They are presented mainly to illustrate the linear method. The potential user is invited to take them and choose his own methods of summarizing the output and of handling fractional replication and missing values.

Appendix A

USE OF THE MULTILINEAR REGRESSION PROGRAM

The data input may be described best by referring to the basic matrix equation

$$XA = Y$$

where X is an n by m matrix, A is a column vector of m elements, and Y is a column vector of n elements. The unknown is the A vector.

Card 1, FORMAT (2I5, F5.0, 54H . . .), has the following parameters:

N (1) = m which is less than 51.

N (2) = n which is less than 1001.

SF = scale factor for Y vector (power of 10).

ID = identification of problem with any legal Hollerith characters.

$m \times n$ = product less than 16,001.

Card 2, FORMAT (12A6), contains identification for each of the m columns in the X matrix. Use as many cards as needed.

Card 3, FORMAT (12F6.0), contains the values of the elements in the Y vector. Use as many cards as needed.

Card 4, FORMAT (12F6.0), contains the values of the elements in the first column of the X matrix. Use as many cards as needed. Repeat this card for each additional column of the X matrix.

When the last column of the X matrix is punched, a complete problem is ready to be processed by the computer. The next card in the data file is Card 1, described above, for another problem; therefore, the above procedure is repeated for the second problem. Any number of problems may be processed.

EXAMPLES OF DATA INPUT

```
    4   41    0  EXAMPLE A
K     X     X2    X3
    3.   22.   39.   54.   65.   76.   85.   90.   95.   99.  102.  103.
  104.  103.  102.  101.   99.   96.   93.   89.   84.   78.   72.   66.
   60.   56.   53.   51.   49.   49.   50.   52.   55.   60.   66.   74.
   84.   95.  108.  121.  137.
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1
     1     2     3     4     5     6     7     8     9    10    11    12
    13    14    15    16    17    18    19    20    21    22    23    24
    25    26    27    28    29    30    31    32    33    34    35    36
    37    38    39    40    41
     1     4     9    16    25    36    49    64    81   100   121   144
   169   196   225   256   289   324   361   400   441   484   529   576
   625   676   729   784   841   900   961  1024  1089  1156  1225  1296
  1369  1444  1521  1600  1681
     1     8    27    64   125   216   343   512   729  1000  1331  1728
  2197  2744  3375  4096  4913  5832  6859  8000  9261 10648 12167 13824
 15625 17576 19683 21952 24389 27000 29791 32768 35937 39304 42875 46656
 50653 54872 59319 64000 68921
    3   28     0  EXAMPLE B
K      LN T  LN T2
  269.  180.  150.  120.   90.   75.   66.   60.   54.   48.   42.   36.
   30.   27.   24.   21.   20.   18.  15.1   14.   12.   10.    8.    6.
    5.    4.    3.   2.5
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1
    0.0.69311.16321.60942.14012.49322.72792.90693.09103.30323.54103.8177
4.12714.31754.49984.76224.85985.06895.41615.57225.87496.24426.68467.2654
7.62568.07098.64829.0119
    0. 0.480 1.353 2.590 4.580 6.216 7.441 8.450 9.55510.91112.53814.575
17.03318.64120.24822.67823.61825.69429.33431.04934.51538.99044.68452.786
58.15065.14074.79281.214
    4     6     0  EXAMPLE IS TAKEN FROM EQUATION 31 OF THE TEXT.
      X2    X3    X4
  108.  111.  120.   99.  115.  117.
    1.    1.    1.    1.    1.    1.
    1.    1.    1.    2.    2.    2.
  300.  400.  600.  350.  550.  650.
 0.001 0.001 0.002 0.001 0.002 0.002
```

MULTILINEAR REGRESSION FORTRAN PROGRAM

```
$IBFTC SIMLIN  LIST,REF,DECK
CMAIN
C     SIMULTANEOUS LINEAR EQUATIONS. MAXIMUM NUMBER OF ELGMENTS IN
C     COEFFICIENT MATRIX IS 16,000.
      COMMON A(16000),Y(1000),D(50,50),YHAT(1000),SS(50),AP(50),X(50),
     1STAT(50),IDENT(9),HEAD(50),RATIO(50),XMEAN(50),XINVE(50,50),M,N,
     2RSDOY,FVFRE,BALM
    1 FORMAT (2I5,F5.0,(9A6))
    2 FORMAT (12A6)
    3 FORMAT (12F6.0)
    5 FORMAT (42H0  IN THE GRAM-SCHMIDT PROCESS THERE WERG ,I3,30H BINAR
     1Y BITS OF ACCURACY LOST.)
    8 FORMAT (1H130X,(9A6)/1H08X,1HY12X,2HY*15X,1HY12X,2HY*15X,1HY12X,2H
     1Y*15X,1HY12X,2HY*/1H )
    9 FORMAT (4(1PE16.5,1PE14.5))
   16 FORMAT (1H130X,(9A6)/1H051X,19HREGRESSION ANALYSIS/20H0  THE MEAN
     1OF Y IS E15.8/39H0  RESIDUAL STANDARD DEVIATION OF Y IS E15.8/29H0
     2  F VALUE FOR REGRESSION IS F10.3/30H0  REGRESSION COEFFICIENTS AR
     3E/1H )
   17 FORMAT (8E15.6)
   18 FORMAT (14H0  X MEANS ARE/1H )
   19 FORMAT (30H0  INVERSE MATRIX ELEMENTS ARE/1H )
   20 FORMAT (1H )
   28 FORMAT (1H130X,(9A6))
    4 READ  (5,1)M,N,SCALE,(IDENT(I),I=1,9)
      READ  (5,2)(HEAD(I),I=1,M)
      READ  (5,3)(Y(I),I=1,N)
      DO 6 J=1,M
      LV = (J - 1)*N + 1
      LA = LV + N - 1
    6 READ  (5,3)(A(I),I=LV,LA)
      SCALE = 10.0**SCALE
      DO 13 I=1,N
   13 Y(I) = Y(I)*SCALE
      CALL LINEQ
      CALL VARFAC
      LV = 50
      DO 10 I=1,N,4
      J = I+3
      IF (N-J) 14,15,15
   14 J = N
   15 IF (50-LV) 11,11,12
   11 WRITE (6,8)(ICENT(K),K=1,9)
      LV = 0
   12 WRITE (6,9)((Y(K),YHAT(K)),K=I,J)
   10 LV = LV+1
      WRITE (6,16)(IDENT(I),I=1,9),XMEAN(1),RSDOY,FVFRE
      LV = 14
      DO 29 I = 1,M,8
      J = I+7
      IF (M-J) 21,22,22
   21 J = M
   22 WRITE (6,17)(X(K),K=I,J)
   29 LV = LV+1
      WRITE (6,18)
      DO 30 I = 2,M,8
      J = I+7
      IF (M-J) 23,24,24
   23 J = M
   24 WRITE (6,17)(XMEAN(K),K=I,J)
   30 LV = LV+1
      I = BALM + 0.5
      WRITE (6,5)I
      WRITE (6,19)
      DO 25 I = 2,M
      IF (LV-51) 27,26,26
   26 LV = 4
      WRITE (6,28)(IDENT(N),N=1,9)
      WRITE (6,19)
   27 WRITE (6,17)(XINVE(I,J),J=2,I)
      WRITE (6,20)
   25 LV = LV+2+((I-2)/8)
      GO TO 4
      END
$IBFTC LINEQ    LIST,REF,DECK
      SUBROUTINE LINEQ
      COMMON A(16000),Y(1000),D(50,50),YHAT(1000),SS(50),AP(50),X(50),
     1STAT(50),IDENT(9),HEAD(50),RATIO(50),XMEAN(50),XINVE(50,50),M,N,
     2RSDOY,FVFRE,BALM
      DIMENSION P(16000)
      EQUIVALENCE(A,P)
      IF (N-1) 11,12,13
   12 X(1) = Y(1)/A(1)
      GO TO 11
   13 DO 2 J=1,M
      DO 2 I=1,M
      D(I,J) = 0.0
      IF (I-J) 2,1,2
    1 D(I,J) = 1.0
    2 CONTINUE
      BALM = 0.0
      YXLO = ALOG(2.0)
      DO 24 J=2,M
      SSBEG = 0.0
      LV = (J-1)*N+1
      LA = LV+N-1
      DO 21 I = LV,LA
   21 SSBEG = SSBEG + P(I)*P(I)
      JM = J-1
      DO 15 K=1,JM
      S = 0.0
      R = 0.0
      LK = (K-1)*N+1
      LM = LK
      DO 4 I = LV,LA
      R = R + P(I)*P(LM)
      S = S + P(LM)*P(LM)
    4 LM = LM + 1
      Q = R/S
      IF (K-1) 20,18,20
   18 XMEAN(J) = Q
   20 LM = LK
      DO 5 I = LV,LA
      P(I) = P(I) - Q*P(LM)
    5 LM = LM + 1
      DO 15 I=1,M
   15 D(I,J) = D(I,J)-Q*D(I,K)
      SSFIN = 0.0
      DO 22 I = LV,LA
   22 SSFIN = SSFIN + P(I)*P:I)
      BALM = BALM + (SSBEG/SSFIN)
   24 CONTINUE
      BALM= (ALOG(BALM)/YXLO)*0.5
      DO 7 J=1,M
      SS(J) = 0.0
      LV = (J-1)*N+1
      LA = LV+N-1
      DO 6 I = LV,LA
    6 SS(J) = SS(J) + P(I)*P(I)
      SS(J) = SQRT(SS(J))
      DO 16 I = LV,LA
   16 P(I) = P(I)/SS(J)
      DO 14 I=1,M
   14 D(I,J) = D(I,J)/SS(J)
      AP(J) = 0.0
      LM = 0
      DO 7 I = LV,LA
      LM = LM + 1
    7 AP(J) = AP(J) + P(I)*Y(LM)
      DO 8 I=1,M
      X(I) = 0.0
      DO 8 J=1,M
    8 X(I) = X(I)+D(I,J)*AP(J)
      DO 17 J=2,M
   17 D(1,J) = 0.0
      DO 9 J=1,M
    9 STAT(J) = AP(J)*AP(J)
      DO 10 I=1,N
      YHAT(I) = 0.0
      DO 10 J=1,M
      LV = (J-1)*N+I
   10 YHAT(I) = YHAT(I) + P(LV)*AP(J)
      DO 19 K=1,M
      DO 19 I=1,M
      XINVE(I,K) = 0 0
      DO 19 J=1,M
   19 XINVE(I,K) = XINVE(I,K)+D(I,J)*D(K,J)
   11 RETURN
      END
$IBFTC VARFAC  LIST,REF,DECK
      SUBROUTINE VARFAC
      COMMON A(16000),Y(1000),D(50,50),YHAT(1000),SS(50),AP(50),X(50),
     1STAT(50),IDENT(9),HEAD(50),RATIO(50),XMEAN(50),XINVE(50,50),M,N,
     2RSDOY,FVFRE,BALM
```

```
  4 FORMAT (1H130X,9A6)
  5 FORMAT (20H0SOURCE OF VARIATION,5X,18HDEGREES OF FREEDOM,6X,15HSUM
   1S OF SQUARES,16X,7HF RATIO/1H )
  6 FORMAT (7X,1A6,18X,I4,14X,E15.8,F22.3)
  7 FORMAT (1H0,6X,8HRESIDUAL,16X,I4,14X,E15.8)
    GMEAN = 0.0
    DO 12 I=1,N
 12 GMEAN = GMEAN+Y(I)
    GN = N
    GMEAN = GMEAN/GN
    XMEAN(1) = GMEAN
    SSTOT = 0.0
    DO 13 I=1,N
    GN = Y(I)-GMEAN
 13 SSTOT = SSTOT+GN*GN
    TTSTAT = 0.0
    DO 14 J=2,M
 14 TTSTAT = TTSTAT+STAT(J)
    SSERR = SSTOT-TTSTAT
    IDF = N-1
    ERRDE = IDF+1-M
    ERRME = SSERR/ERRDE
    RSDOY = SQRT(ERRME)
    GN = M-1
    FVFRE = (ERRDE*TTSTAT)/(GN*SSERR)
    DO 15 I=2,M
 15 RATIO(I) = STAT(I)/ERRME
    WRITE (6,4)(IDENT(I),I=1,9)
    WRITE (6,5)
    K = 1
    DO 16 I=2,M
 16 WRITE (6,6)HEAD(I),K,STAT(I),RATIO(I)
    K = ERRDE
    WRITE (6,7)K,SSERR
    DATA Q000HL/6HTOTAL /
    HEAD(1) = Q000HL
    WRITE (6,6)HEAD(1),IDF,SSTOT
    RETURN
    END
```

Appendix B

USE OF THE ORTHOGONAL POLYNOMIAL AND CURVILINEAR REGRESSION PROGRAM

The data input may be described by referring to the curvilinear equation

$$y_i = \sum_{j=0}^{m} a_j x_i^j \qquad i = 1, 2, \ldots, n$$

where a_j are the unknowns.

Card 1, FORMAT (13I1, I2, 3I3), has the following parameters:

L (1) = not used.

L (2) = 1 to read in new x values or 0 to use previous values.

L (3) = 1 to print out transformation matrix (otherwise 0).

L (4) = 1 to print out orthogonal polynomials (otherwise 0).

L (5) = 1 to punch out orthogonal polynomials (otherwise 0).

L (6) = 1 to read in new y values or 0 to use previous values.

L (7) = 1 to print out regression coefficients for orthogonal polynomials (otherwise 0).

L (8) = 1 to print out conventional regression coefficients (otherwise 0).

L (9) = 1 to print out correlation coefficients (otherwise 0).

L (10) = 1 to print out y values predicted from regression (otherwise 0).

L (11) = not used.

L (12) = not used.

L (13) = not used.

L (14) = m which is less than 20. All degrees of fit from 1 to m are computed.

L (15) = number (n) of points. It is less than 1,000 but must be greater than m.

L (16) = scale factor for x values (power of 10).

L (17) = scale factor for y values (power of 10).

For large problems, x and y values should be scaled so that the data are close to unity.

Card 2, FORMAT 72H . . .), identifies the problem with any legal Hollerith characters. The character in the first column enters the carriage control channel.

Card 3, FORMAT (12F6.0), contains the x values. Use as many cards as needed. This card must not be in the data file if L (2) = 0, in which case the x values from the previous problem are used. (L (2) = 1 for the first problem.)

Card 4, FORMAT (12F6.0), contains the y values. Use as many cards as needed. This card must not be in the data file if L (6) = 0, in which case the y values from the previous problem are used. (L (6) = 1 for the first problem.)

With the completion of Card 4, the next card in the data file is Card 1, described above, for the next problem; thus, the above process is repeated for another problem. Any number of problems may be processed.

The program is coded in single precision, and thus large differencing errors may appear for high degrees of fit.

EXAMPLES OF DATA INPUT

```
 1   1 111    1  4 -6 -4
1  EXAMPLE ONE
  3002  2915  2832  2754
 27993 25224 22989 20569
     1 111    1  4 -6 -4
1  EXAMPLE TWO
28195 25623 23201 20792
 1     111    1  4 -6 -4
1  EXAMPLE THREE
  3007  2915  2826  2743
 111 1           4  5
1  EXAMPLE FOUR
    -2    -1     0     1     2
     1     2     3     4     5
 1   1 111    4 10 -5 -3
1  EXAMPLE FIVE
  2525  2828  3147  3586  4198  4456  5757  6688  7662  8704
  1000  1099  1144  1188  1228  1253  1304  1339  1374  1408
```

ORTHOGONAL POLYNOMIAL AND CURVILINEAR REGRESSION FORTRAN PROGRAM

```
$IBFTC CURVLN  LIST,REF,DECK
      COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
    1 FORMAT(13I1,I2,3I3,F6.6)
    2 FORMAT(54H1BAD CONTROL CARD OR MISCOUNT OF INPUT  QUIT THIS JOB.)
   10 READ  (5,1)(L(I),I = 1,17)
    9 M = L(14)
      N = L(15)
      IF (N-1-M) 1111,13,13
   13 DO 14 K = 1,13
      IF (L(K) -1) 14,14,1111
   14 CONTINUE
      IF (M-19) 15,15,1111
   15 CALL ID
      IF (L(2)) 17,17,16
   16 CALL XFILE
   17 IF (L(3)) 19,19,18
   18 CALL DETMAT
   19 IF (L(4))21,21,20
   20 CALL WRITEP
   21 IF (L(5))26,26,24
   24 CALL PUNCHP
   26 IF (L(6)) 28,28,27
   27 CALL YFILE
   28 IF(L(7))30,30,29
   2° CALL OUTB
   30 IF(L(8))32,32,31
   31 CALL OUTA
   32 IF(L(9))34,34,33
   33 CALL OUTR
      CALL OUTFR
   34 IF(L(10))10,10,35
   35 CALL OUT
      GO TO 10
 1111 WRITE (6,2)
      CALL EXIT
      END
$IBFTC OUTFR   LIST,REF,DECK
      SUBROUTINE OUTFR
      COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
      COMMON SSY
      DIMENSION  FA(20), FR(20), Q(20),R(20),S(20)
    1 FORMAT (51H0F-RATIOS (INDIVIDUAL) D.F.(NUM)=1, D.F.(DEN)=N-1-J)
    2 FORMAT(65H0F-RATIOS (ACCUMULATIVE) D.F.(NUM)=J, D.F.(DEN)=N-1-J, J
     1=0,1.....,I2)
     3 FORMAT (1H )
     4 FORMAT(1H0,10F11.3)
     7 FORMAT (25H0RESIDUAL VARIATION IN Y.)
     8 FORMAT (1H0,7E15.6)
     9 FORMAT (36H0RESIDUAL STANDARD DEVIATION OF FIT.)
       QQ = N-2
       MP = M+1
       DO 5    J = 2,MP
     5 Q(J) = B(J)**2
       D = SSY
       A = 0.0
       DO 6  J = 2,MP
       S = N - J
       T = J - 1
       QQ = S/T
       A = A + Q(J)
       D = D - Q(J)
       R(J) = D
       FA(J) = Q(J)/D *QQ
     6 FB(J) = A/D*QQ
       R(1) = SSY
       DO 11   J = 1,MP
       X = N-J
    11 S(J) = SQRT(R(J)/X)
       WRITE (6,3)
       WRITE (6,1)
       WRITE (6,4)(FA(J), J = 1,MP)
       WRITE (6,3)
       WRITE (6,2)M
       WRITE (6,4)(FB(J), J = 1,MP)
       WRITE (6,3)
       WRITE (6,7)
       WRITE (6,8)(R(J), J = 1,MP)
       WRITE (6,3)
       WRITE (6,9)
       WRITE (6,8)(S(J), J = 1,MP)
       RETURN
       END
$IBFTC OUTR     LIST,REF,DECK
       SUBROUTINE OUTR
       COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
       COMMON SSY
       DIMENSION R(20),V(20)
     1 FORMAT (26HOCORRELATION COEFFICIENTS.)
     2 FORMAT (50HOY VARIATION ACCOUNTED FOR 3Y J TH POLYNOMIAL SET )
     3 FORMAT (1H )
     4 FORMAT(1H ,F9.5,11F10.5)
     5 FORMAT(1H ,7E15.6)
       WRITE (6,1)
       WRITE (6,3)
       Q = 1.0/SQRT(SSY)
       MP = M + 1
       DO 6 J = 2,MP
       V(J) =          B(J)**2
     6 R(J) = B(J)*Q
       WRITF (6,4)(R(J),J = 1, MP)
       WRITE (6,3)
       WRITE (6,2)
       WRITE (6,3)
       WRITE (6,5)(V(J),J = 1,MP)
       RETURN
       END
$IBFTC ID       LIST,REF,DECK
       SUBROUTINE ID
       COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
       COMMON SSY
     1 FORMAT(65H1ORTHOGONAL POLYNOMIAL  AND CURVILINEAR REGRESSION ROUTI
      1NE 131-22,I12,11I1)
     2 FORMAT (72H
      1                     )
     5 FORMAT (1H )
     3 FORMAT(13H0DFGREE FIT =,I3,5H, N =,I4)
     4 READ  (5,2)
       WRITE (6,1)(L(I),I=1,12)
       WRITE (6,2)
       WRITE (6,3)M,N
       WRITF (6,5)
       RETURN
       END
$IBFTC DETMAT  LIST,REF,DECK
       SUBROUTINE DETMAT
       COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
      COMMON SSY
    1 FORMAT (24H0MATRIX OF DETERMINATION)
    2 FORMAT (1H )
    3 FORMAT (1H ,7E15.6)
    6 FORMAT ( 1H1)
    4 WRITF (6,1)
      WRITE (6,2)
      MP = M + 1
      DO 5  J = 1,MP
      WRITE (6,3)(XB(K,J),K = 1,J)
    5 WRITE (6,2)
      WRITE (6,6)
      RETURN
      END
$IBFTC WRITEP  LIST,REF,DECK
      SUBROUTINE WRITEP
      COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
      COMMON SSY
    1 FORMAT (23H0ORTHOGONAL POLYNOMIALS)
    3 FORMAT (5H0J = ,I2)
    4 FORMAT (1H ,E20.6,6E15.6)
    5 WRITE (6,1)
      MP = M + 1
      DO 6 J = 2,MP
      K = J - 1
    2 WRITE (6,3)K
    6 WRITE (6,4)(P(I,J) , I = 1,N)
      RETURN
      END
$IBFTC OUTB    LIST,REF,DECK
      SUBROUTINE OUTB
      COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
      COMMON SSY
    1 FORMAT (52H0REGRESSION COEFFICIENTS FOR ORTHOGONAL POLYNOMIALS    )
    2 FORMAT (1H )
    4 FORMAT (1H ,7E15.6)
    3 WRITE (6,1)
      WRITE (6,2)
      MP = M + 1
      WRITE (6,4)(B(J), J = 1,MP)
      WRITE (6,2)
      RETURN
      END
$IBFTC XFILE   LIST,REF,DECK
      SUBROUTINE XFILE
      COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
      COMMON SSY
      DIMENSION   R2(20)
    1 FORMAT (12F6.0)
    2 READ  (5,1)(P(I,2), I=1,N)
      MP = M + 1
      DO 3  I = 1:N
    3 P(I,1) = 1.0
      IF (3-MP)  32,32,36
   32 DO 33 J = 3,MP
      DO 33  I = 1,N
   33 P(I,J) = P(I,2)**(J-1)
   36 DO 4  J = 1,MP
      DO 4  I = 1,MP
      XB(I,J) = 0.0
      IF (I-J)  4,5,4
    5 XB(I,J) = 10.0**(L(16)*(1-J))
    4 CONTINUE
      DO 7  J = 2,MP
      JM = J-1
      DO 7  K = 1,JM
      S = 0.0
      R = 0.0
      DO 6  I = 1,N
      R = R + P(I,J)*P(I,K)
    6 S = S + P(I,K)**2
      C = R/S
      DO 8  I = 1,MP
    8 XB(I,J) = XB(I,J) - C*XB(I,K)
      DO 7  I = 1,N
    7 P(I,J) = P(I,J) - C*P(I,K)
      DO 9  J = 1,MP
      SS(J) = 0.0
      DO 9  I = 1,N
    9 SS(J) = SS(J) + P(I,J)**2
      DO 11  J = 1,MP
      S = 1.0/SQRT(SS(J))
      DO 10  I = 1,MP
   10 XB(I,J) = XB(I,J)*S
      DO 11  I = 1,N
   11 P(I,J) = P(I,J)*S
      RETURN
      END
$IBFTC YFILE    LIST,REF,DECK
      SUBROUTINE YFILE
      COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
      COMMON SSY
      DIMENSION YY(999)
    7 FORMAT(42H1REGRESSION ANALYSIS , SUM OF SQUARES Y = ,E15.6)
    2 FORMAT (1H )
    3 FORMAT (12F6.0)
      READ  (5,3)(Y(I),I=1,N)
      Q = 10.0**L(17)
      DO 4 I = 1,N
      Y(I) = Q * Y(I)
    4 YY(I) = Y(I)
      MP = M + 1
      DO 9   J = 1,MP
      B(J) = 0
      DO 5 I = 1,N
    5 B(J) = B(J) + YY(I)*P(I,J)
      DO 9   I = 1,N
    9 YY(I) = YY(I) - B(J)*P(I,J)
      SSY = 0
      YB = B(1)*XB(1,1)
      DO   6   I = 1,N

    6 SSY = SSY + (Y(I) - YB) **2
      WRITE (6,7)SSY
      WRITE (6,2)
      RETURN
      END
$IBFTC OUTA     LIST,REF,DECK
      SUBROUTINE OUTA
      COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
      COMMON SSY
      DIMENSION  A(20,20 )
    1 FORMAT (38H0CONVENTIONAL REGRESSION COEFFICIENTS.)
    2 FORMAT (14H0DEGREE FIT = ,I2)
    3 FORMAT (1H ,7E15.6)
    7 FORMAT (1H1)
      WRITE (6,1)
      MP = M+1
      A(1,1) = B(1)*XB(1,1)
      DO 6 J = 2,MP
      DO 4  K = 1,J
      4 A(J,K) = A(J-1,K) + XB(K,J)*B(J)
      6 CONTINUE
        DO 5   J = 1,MP
        KM = J-1
        WRITE (6,2)KM
      5 WRITE (6,3)(A(J,K), K = 1,J)
        WRITE (6,7)
        RETURN
        END
$IBFTC OUT      LIST,REF,DECK
        SUBROUTINE OUT
        DIMENSION T(10)
        COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
        COMMON SSY
      1 FORMAT (39H1Y AND PREDICTED Y,S FROM M DEGREE FIT.)
      2 FORMAT  (1H )
      3 FORMAT (1H ,F19.0,F10.0,F20.0,F10.0,F20.0,F10.0,F20.0,F10.0)
      4 FORMAT (13H0(NOT SCALED))
        WRITE (6,1)
        WRITE (6,4)
        WRITE (6,2)
        Q = 1.0/10.0**L(17)
        MP = M+1
        KL = 8
        DO 6  I = 1,N,4
        IF (N-I) 9,8,9
      8 KL = 2
      9 T(1) = Y(I)*Q
        T(2) = 0.0
        IF (N-I-1)  11,10,11
     10 KL = 4
     11 T(3) = Y(I+1)*Q
        T(4) = 0.0
        IF (N-I-2)  13,12,13
     12 KL = 6
     13 T(5) = Y(I+2)*Q
        T(6) = 0.0
        T(7) = Y(I+3)*Q
        T(8) = 0.0
        DO 5    J = 1,MP
        T(2) = T(2) + P(I,J)*B(J)*Q
        T(4) = T(4) + P(I+1,J)*B(J)*Q
        T(6) = T(6) + P(I+2,J)*B(J)*Q
      5 T(8) = T(8) + P(I+3,J)*B(J)*Q
      6 WRITE (6,3)(T(K), K = 1,8)
        RETURN
        END
$IBFTC PUNCHP  LIST,REF,DECK
        SUBROUTINE PUNCHP
        COMMON P(999,20),M,N,L(17),Y(999), B(20), SS(20), XB(20,20), TOL
        COMMON SSY
      1 FORMAT (23H0ORTHOGONAL POLYNOMIALS)
      3 FORMAT (5H0J = ,I2)
      4 FORMAT (1H ,E15.6,4E15.6)
      5 WRITE (3,1)
        MP = M + 1
        DO 6 J = 1,MP
        K = J - 1
      2 PUNCH 3, K
      6 PUNCH 4, (P(I,J), I = 1,N)
        RETURN
        END
```

Appendix C

USE OF THE FACTORIAL ANALYSIS PROGRAM

The data input may be divided into two parts. The first part is the orthonormal tensors, and the second part is the measured data points which are to be arranged in a factorial design. Using the notation of the text, the orthonormal tensors are designated as

$$\underline{P}_{s_i}^{j_i} \qquad \begin{array}{l} 2 \leq i \leq 9 \\ 1 \leq s \leq i \\ 1 \leq j \leq i \end{array}$$

The index i designates a tensor which is congruent to a square matrix of order i. The index s denotes a normal vector of the ith tensor, and the index j denotes a component of the sth normal vector. For an ith tensor, there are i normal vectors and i components for each normal vector. Thus there are i^2 elements.

Card 1, FORMAT (I1), gives the order, i, of the tensor which appears on the next card.

Card 2, FORMAT (5X, 5E 15.7), contains the i values of the first normal vector of the ith tensor. If i is greater than 5, use another card with the same format. Repeat this card for each additional normal vector of the ith tensor.

When the ith normal vector is punched, repeat the above cards for another tensor. The process is terminated when i is equal to zero. These tensors (orthogonal polynomials) may be obtained in the proper format from the program in Appendix B.

The second part of the data input is for the measured data points of a factorial design.

Card 1, FORMAT (2I5, F5.0, 57H . . .) has the following parameters:

N (1) = number of factors in the design. It must be less than 15. Replication is not considered to be a factor.

N (2) = number of data points in the design. It must be less than 21,001.

SF = scale factor for the data points (power of 10).

ID = identification of problem with any legal Hollerith characters.

Card 2, FORMAT (4X, 15I4), has the following parameters:

L (1) = number of replications.

L (2) = number of levels for first factor.

L (3) = number of levels for second factor.

. . .

L (15) = number of levels for fourteenth factor.

If a minus sign is punched in front of the number of levels, that factor is considered to be quantitative. Otherwise, the factor is considered to be qualitative. Each distinct number punched, for the number of levels of a factor, presupposes that an orthogonal tensor of that order has been included in the first part of the data file.

Card 3, FORMAT (12F6.0), contains the measured data points of the factorial design. Use as many cards as needed. The data points must be ordered by the level indices of the replicate and of the factors, according to the hybrid-base number system (see text), where the highest-order index is for replication, the next order index is for the first factor, etc., until the lowest order index is for the last factor of the design.

When all the data points have been punched, one factorial design has been completed. Any number of factorial designs may be processed by repeating the above cards for the second part of the data input. The first part of the data input is not repeated for each factorial design. If different tensors are to be used for a design, a new data run must be made on the computer with both parts of the data input present. A limitation on the use of the program is that two factors having the same number of levels cannot be analyzed with two different tensors.

EXAMPLES OF DATA INPUT

```
2
         7.0710678E-01  7.0710678E-01
        -7.0710678E-01  7.0710678E-01
3
         5.7735027F-01  5.7735027E-01  5.7735027E-01
        -7.0710678F-01  0.             7.0710678E-01
         4.0824829F-01 -8.1649658E-01  4.0824829E-01
4
         5.0000000F-01  5.0000000F-01  5.0000000F-01  5.0000000E-01
        -6.7082039F-01 -2.2360680E-01  2.2360680E-01  6.7082039E-01
         5.0000000F-01 -5.0000000E-01 -5.0000000F-01  5.0000000E-01
        -2.2360680E-01  6.7082039F-01 -6.7082039E-01  2.2360680E-01
5
         4.4721360E-01  4.4721360E-01  4.4721360E-01  4.4721360E-01  4.4721360E-01
        -6.3245553F-01 -3.1622777E-01  0.             3.1622777E-01  6.3245553E-01
         5.3452248F-01 -2.6726124E-01 -5.3452248E-01 -2.6726124E-01  5.3452248E-01
        -3.1622777F-01  6.3245553F-01  0.            -6.3245553E-01  3.1622777E-01
         1.1952286E-01 -4.7809144E-01  7.1713717E-01 -4.7809144E-01  1.1952286E-01
6
         4.0824829F-01  4.0824829F-01  4.0824829E-01  4.0824829E-01  4.0824829E-01
         4.0824829E-01
        -5.9761430F-01 -3.5856858E-01 -1.1952286F-01  1.1952286E-01  3.5856858E-01
         5.9761430E-01
         5.4554473E-01 -1.0910895E-01 -4.3643578E-01 -4.3643578E-01 -1.0910895E-01
         5.4554473E-01
        -3.7267800F-01  5.2174919E-01  2.9814240E-01 -2.9814240E-01 -5.2174919E-01
         3.7267800E-01
         1.8898224F-01 -5.6694671E-01  3.7796447E-01  3.7796447E-01 -5.6694671E-01
         1.8898224E-01
        -6.2994079F-02  3.1497039E-01 -6.2994079E-01  6.2994079E-01 -3.1497039E-01
         6.2994079E-02
7
         3.7796447F-01  3.7796447F-01  3.7796447E-01  3.7796447E-01  3.7796447E-01
         3.7796447E-01  3.7796447E-01
        -5.6694671E-01 -3.7796447E-01 -1.8898224E-01  0.             1.8898224E-01
         3.7796447F-01  5.6694671E-01
         5.4554473E-01  0.            -3.2732684E-01 -4.3643578E-01 -3.2732684E-01
         0.             5.4554473E-01
        -4.0824829E-01  4.0824829E-01  4.0824829E-01  0.            -4.0824829E-01
        -4.0824829E-01  4.0824829E-01
         2.4174689F-01 -5.6407607E-01  8.0582296E-02  4.8349378E-01  8.0582296E-02
        -5.6407607F-01  2.4174689E-01
        -1.0910895F-01  4.3643578E-01 -5.4554473E-01  0.             5.4554473E-01
        -4.3643578F-01  1.0910895E-01
         3.2897585E-02 -1.9738551E-01  4.9346377E-01 -6.5797169E-01  4.9346377E-01
        -1.9738551F-01  3.2897585E-02
```

```
8
      3.5355339E-01  3.5355339E-01  3.5355339E-01  3.5355339E-01  3.5355339E-01
      3.5955339F-01  3.5355339E-01  3.5355339E-01
     -5.4006172E-01 -3.8575837E-01 -2.3145502E-01 -7.7151675E-02  7.7151675E-02
      2.3145502E-01  3.8575837E-01  5.4006172E-01
      5.4006172E-01  7.7151675E-02 -2.3145502E-01 -3.8575837E-01 -3.8575837E-01
     -2.3145502E-01  7.7151675E-02  5.4006172E-01
     -4.3082022E-01  3.0772873E-01  4.3082022F-01  1.8463724E-01 -1.8463724E-01
     -4.3082022F-01 -3.0772873E-01  4.3082022F-01
     -2.8203804E-01 -5.2378493E-01 -1.2087344E-01  3.6262033E-01  3.6262033E-01
     -1.2087344E-01 -5.2378493E-01  2.8203804E-01
     -1.4978617E-01  4.9215457E-01 -3.6376642E-01 -3.2097037E-01  3.2097037E-01
      3.6376642E-01 -4.9215457E-01  1.4978617E-01
      6.1545745E-02 -3.0772873E-01  5.5391171E-01 -3.0772873E-01 -3.0772873E-01

      5.5391171E-01 -3.0772873E-01  6.1545745E-02
     -1.7069719E-02  1.1948803E-01 -3.5846409E-01  5.9744015E-01 -5.9744015E-01
      3.5846409E-01 -1.1948803E-01  1.7069719E-02
9
      3.3333333E-01  3.3333333F-01  3.3333333F-01  3.3333333E-01  3.3333333E-01
      3.3333333E-01  3.3333333E-01  3.3333333E-01  3.3333333E-01
     -5.1639778E-01 -3.8729833E-01 -2.5819889E-01 -1.2909944E-01  0.
      1.2909944E-01  2.5819889E-01  3.8729833E-01  5.1639778E-01
      5.3181602E-01  1.3295401E-01 -1.5194744E-01 -3.2288830E-01 -3.7986859E-01
     -3.2288830E-01 -1.5194744E-01  1.3295401F-01  5.3181602E-01
     -4.4494921E-01  2.2247460E-01  4.1316712F-01  2.8603878E-01  0.
     -2.8603878E-01 -4.1316712E-01 -2.2247460E-01  4.4494921E-01
      3.1289311E-01 -4.6933966E-01 -2.4584459E-01  2.0114557E-01  4.0229114E-01
      2.0114557E-01 -2.4584459E-01 -4.6933966E-01  3.1289311E-01
     -1.8490007E-01  5.0847518E-01 -1.8490007E-01 -4.1602515E-01  0.
      4.1602515E-01  1.8490007E-01 -5.0847518E-01  1.8490007E-01
      8.9893315E-02 -3.8204659E-01  4.9441323E-01  2.2473329E-02 -4.4946657E-01
      2.2473329E-02  4.9441323E-01 -3.8204659F-01  8.9893315E-02
     -3.4139437E-02  2.0483662E-01 -4.7795212F-01  4.7795212E-01  0.
     -4.7795212E-01  4.7795212E-01 -2.0483662E-01  3.4139437E-02
      8.8147648F-03 -7.0518118E-02  2.4681341E-01 -4.9362683E-01  6.1703353E-01
     -4.9362683E-01  2.4681341E-01 -7.0518118E-02  8.8147648E-03
0
    4   48   -3  EXAMPLE NUMBER ONE
        1    2    2   -3   -4
  2753  2007  1959  3010  2913  2257  1973  2563  2666  2160  1803  2520
  2649  2225  1990  2582  2542  2210  2105  2764  2503  2215  2064  2843
  2438  2312  2086  2405  2766  2419  2325  2644  2608  2203  2048  2544
  2335  2225  1944  2838  2579  2003  2218  2494  2655  2230  2084  2712
    4  189    0  EXAMPLE NUMBER TWO
        1   -3    7   -3    3
 1.205 1.206 1.201 1.205 1.205 1.204 1.210 1.208 1.210 1.200 1.199 1.200
 1.200 1.203 1.201 1.200 1.206 1.203 1.205 1.206 1.205 1.205 1.205 1.210
 1.205 1.210 1.212 1.150 1.148 1.150 1.150 1.151 1.155 1.155 1.153 1.156
 1.145 1.138 1.139 1.140 1.139 1.139 1.140 1.140 1.140 1.160 1.159 1.159
 1.165 1.165 1.161 1.165 1.162 1.163 1.050 1.056 1.050 1.055 1.053 1.059
 1.045 1.052 1.053 1.050 1.043 1.043 1.050 1.042 1.041 1.035 1.043 1.045
 1.050 1.049 1.050 1.050 1.050 1.049 1.035 1.050 1.050 1.320 1.318 1.319
 1.315 1.319 1.319 1.305 1.319 1.320 1.330 1.326 1.314 1.320 1.322 1.325
 1.330 1.328 1.325 1.320 1.319 1.322 1.315 1.312 1.320 1.320 1.312 1.319
 1.390 1.385 1.383 1.385 1.384 1.388 1.390 1.385 1.390 1.395 1.397 1.393
 1.395 1.395 1.397 1.395 1.398 1.395 1.400 1.395 1.391 1.395 1.398 1.391
 1.395 1.392 1.391 1.180 1.185 1.180 1.180 1.180 1.180 1.185 1.182 1.182
 1.165 1.159 1.161 1.165 1.163 1.161 1.165 1.161 1.163 1.180 1.175 1.175
 1.175 1.178 1.178 1.180 1.179 1.179 1.990 1.988 1.990 1.990 1.990 1.993
 2.000 1.994 1.995 2.020 2.012 2.015 2.015 2.016 2.016 2.020 2.015 2.018
 2.020 2.021 2.023 2.020 2.019 2.025 2.025 2.022 2.025
```

FACTORIAL ANALYSIS FORTRAN PROGRAM

```
$IBFTC MAIN     LIST,REF,DECK
CMAIN
C      ANALYSIS OF VARIANCE
       COMMON P(9,9,9),INDEX(15),KOG(15),LEV(15),LUMP(15),D(21000),LIP(15
      1),VECTOR(9),HEAD(10),HOLER(22),LEVEL,M,N,MF
    1 FORMAT (2I5,F5.0,9A6,A3)
    2 FORMAT (4HO    ,15I4)
    3 FORMAT (12F6.0)
    4 FORMAT (80HO       REP X1  X2  X3  X4  X5  X6  X7  X8  X9  X10 X11 X
     112 X13 X14 SUM OF SQUARES/1H )
    5 FORMAT (4H     ,15I4,E16.8)
   15 FORMAT (1H1,30X,10A6)
   16 FORMAT (11 HOTHERE ARE ,I2,10H FACTORS, ,I5,42H DATA POINTS OF WHI
     1CH THE SCALE FACTOR IS ,F5.0)
      CALL POLY
    6 READ  (5,1)M,N,SCALE,(HEAD(I),I=1,9),HEAD(10)
      M = M+1
      READ  (5,2)(LEV(K), K = 1,M)
      DO 23 K=1,M
      IF (LEV(K))21,13,22
   21 LIN(K) = 1
      LEV(K) = -LEV(K)
      GO TO 23
   13 LEV (K) = 1
   22 LIN (K) = 0
   23 CONTINUE
   14 IF (LEV(1)-1) 18,18,20
   18 MF = 2
      GO TO 19
   20 MF = 1
   19 SC = SCALE
      LUMP(M) = 1
      DO 7  I = 2,M
      J = M-I+1
    7 LUMP(J) = LUMP(J+1)*LEV(J+1)
      READ  (5,3)(D(I), I = 1,N)
      SCALE = 10.0**SCALE
      DO 9 I = 1,N
    9 D(I) = D(I)*SCALE
      DO 17 K=1,15
   17 KOG(K) = 0
      WRITE (6,15)(HEAD(I),I = 1,10)
      K = M-1
      WRITE (6,16)K,N,SC
      CALL MEANS
      CALL ANLSIS
      DO 12 I=1,N
   12 D(I) = D(I)**2
      LEVEL = 1
      LV = 50
      DO 8   I = 1,N
      IF (50-LV) 10,10,24
   10 WRITE (6,15)(HEAD(J),J - 1,10)
      WRITE (6,16)K,N,SC
      WRITE (6,2)(LEV(K), K = 1,M)
      WRITE (6,4)
      LV =0
   24 WRITE (6,5)(KOG(K), K = 1,15), D(I)
      CALL COUNT (16)
    8 LV = LV+1
      CALL ANOVA
      GO TO 6
      END
```

```
$IBFTC POLY     LIST,REF,DECK
      SUBROUTINE POLY
      COMMON P(9,9,9),INDEX(15),KOG(15),LEV(15),LUMP(15),D(21000),LIN(15
     1),VECTOR(9),HEAD(10),HOLER(22),LEVEL,M,N,MF
    1 FORMAT  (I1)
    2 FORMAT (1H , E19.6, 4E15.6)
    3 READ  (5,1)N
      IF (N) 6,6,4
    4 DO 5   J = 1,N
    5 READ  (5,2)(P(N,J,I), I = 1,N)
      GO TO 3
      DATA Q000HL/6HREP    /
    6 HOLER(1) = Q000HL
      DATA Q001HL/6HX1     /
      HOLER(2) = Q001HL
      DATA Q002HL/6HX2     /
      HOLER(3) = Q002HL
      DATA Q003HL/6HX3     /
      HOLER(4) = Q003HL
      DATA Q004HL/6HX4     /
      HOLER(5) = Q004HL
      DATA Q005HL/6HX5     /
      HOLER(6) = Q005HL
      DATA Q006HL/6HX6     /
      HOLER(7) = Q006HL
      DATA Q007HL/6HX7     /
      HOLER(8) = Q007HL
      DATA Q008HL/6HX8     /
      HOLER(9) = Q008HL
      DATA Q009HL/6HX9     /
      HOLER(10) = Q009HL
      DATA Q010HL/6HX10    /
      HOLER(11) = Q010HL
      DATA Q011HL/6HX11    /
      HOLER(12) = Q011HL
      DATA Q012HL/6HX12    /
      HOLER(13) = Q012HL
      DATA Q013HL/6HX13    /
      HOLER(14) = Q013HL
      DATA Q014HL/6HX14    /
      HOLER(15) = Q014HL
      DATA Q015HL/6HLIN    /
      HOLER(16) = Q015HL
      DATA Q016HL/6HQUA    /
      HOLER(17) = Q016HL
      DATA Q017HL/6HCUB    /
      HOLER(18) = Q017HL
      DATA Q018HL/6H,      /
      HOLER(19) = Q018HL
      DATA Q019HL/6H(      /
      HOLER(20) = Q019HL
      DATA Q020HL/6H),     /
      HOLER(21) = Q020HL
      DATA Q021HL/6H      ./
      HOLER(22) = Q021HL
      RETURN
      END
$IBFTC ANLSIS  LIST,REF,DECK
      SUBROUTINE ANLSIS
      COMMON P(9,9,9),INDEX(15),KOG(15),LEV(15),LUMP(15),D(21000),LIN(15
     1),VECTOR(9),HEAD(10),HOLER(22),LEVEL,M,N,MF
      DO 3   J = MF,M
      DO 1   K = 1,15
    1 KOG(K) = 0.0
      LEVEL = LEV(J)
      DO 3   I = 1,N,LEVEL
      CALL COUNT (J)
      DO 2   K = 1,LEVEL
      IN = INDEX(K)
    2 VECTOR(K) = D(IN)
      DO 3   K = 1,LEVEL
      IN = INDEX(K)
      D(IN) = 0.0
      DO 3   L = 1,LEVEL
    3 D(IN) = D(IN) + VECTOR(L)*P(LEVEL,K,L)
      RETURN
      END
$IBFTC COUNT   LIST,REF,DECK
      SUBROUTINE COUNT (J)
      COMMON P(9,9,9),INDEX(15),KOG(15),LEV(15),LUMP(15),D(21000),LIN(15
     1),VECTOR(9),HEAD(10),HOLER(22),LEVEL,M,N,MF
    1 INDEX(1) = 1
      DO 2   I = 1,M
    2 INDEX(1) = INDEX(1) + LUMP(I)*KOG(I)
      DO 3   I = 2,LEVEL
    3 INDEX(I) = INDEX(I-1) + LUMP(J)
      DO 5   I = 1,M
      L = M-I+1
      IF (L-J) 4,5,4
    4 KOG(L) = KOG(L) + 1
      IF (KOG(L)-LEV(L)) 6,5,5
    5 KOG(L) = 0
    6 RETURN
      END
$IBFTC ANOVA   LIST,REF,DECK
      SUBROUTINE ANOVA
      COMMON P(9,9,9),INDEX(15),KOG(15),LEV(15),LUMP(15),D(21000),LIN(15
     1),VECTOR(9),HEAD(10),HOLER(22),LEVEL,M,N,MF
      DIMENSION SS(15),SSS(15)
   41 FORMAT (1H130X,10A6/20H0SOURCE OF VARIATION6X,18HDEGREES OF FREEDO
     1M7X,14HSUM OF SQUARES20X,7HF RATIO/1H )
   42 FORMAT (1H )
   43 FORMAT (8X,2A3,I22,E29.8,F26.3)
   44 FORMAT (1H06X,8HRESIDUALI21,E29.8)
   45 FORMAT (7X,5HTOTALI24,E29.8)
   46 FORMAT (1H I2,34H WAY INTERACTION SUMS OF SQUARES =E17.8)
   47 FORMAT (1H I2,45H WAY AND HIGHER INTERACTION SUMS OF SQUARES =E17.
     18)
   48 FORMAT (1H1)
   63 FORMAT (1H0/1H06X,30HRESIDUAL STANDARD DEVIATION = E15.8)
   64 FORMAT (8X,2A3,A1,2A3,I15,E29.8,F26.3)
      WRITE (6,41)(HEAD(J),J=1,10)
      LV = 0
      DO 1 I=1,15
      SSS(I)=0.0
      SS(I)=0.0
    1 KOG(I)=0
      ERSS=0.0
      ERDF=0.0
      TSS=0.0
      DO 6 I=2,N
      CALL COUNT(16)
      K=0
      DO 3 J=1,M
      IF (KOG(J)) 3,3,2
    2 K=K+1
    3 CONTINUE
      SS(K)=SS(K)+D(I)
      TSS=TSS+D(I)
      IF (K-2) 4,31,5
   31 IF (KOG(1)) 4,4,5
```

```
 4    DO 7 J=2,M
      IF (KOG(J)-3) 7,7,21
 21   IF (LIN(J)) 7,7,5
 7    CONTINUE
      GO TO 6
 5    ERSS=ERSS+D(I)
      ERDF=ERDF+1.0
 6    CONTINUE
      IF (MF-1) 27,26,27
 26   K=1
      GO TO 25
 27   K = 2
 25   DO 12 KA=K,M
      DO 8 I=1,15
 8    KOG (I)=0
      IN=1
      DF = 0.0
      ALP = 0.0
      DO 9 I=2,N
      CALL COUNT(16)
      IF (KOG(KA)-IN) 9,10,12
 10   IF (LIN(KA)) 32,32,33
 32   DF = DF + 1.0
      ALP = ALP + D(I)
      GO TO 34
 33   DF = 1.0
      ALP = D(I)
      IM = IN + 15
 36   RATIO = (ERDF*ALP)/(ERSS*DF)
      J = DF
      WRITE (6,43)HOLER(KA),HOLER(IM),J,ALP,RATIO
      LV = LV+1
      IF (IN-11) 34,12,12
 34   IN = IN +1
      IF (LEV(KA)-IN) 12,35,11
 35   IF (LIN(KA)) 37,37,12
 37   IN = 11
      IM = 22
      GO TO 36
 11   IF (LIN(KA)) 9,9,38
 38   IF (IN-3)9,9,12
 9    CONTINUE
 12   CONTINUE
      K=M-1
      DO 18 KA=2,K
      KB=KA+1
      DO 18 KC=KB,M
      IM=1
 55   IN=1
      DO 13 I=1,15
 13   KOG(I)=0
      DF = 0.0
      ALP = 0.0
      DO 14 I=2,N
      CALL COUNT (16)
      IF (KOG(KA)-IN) 14,15,18
 15   IF (KOG(KC)-IM) 14,16,18
 16   IF (LIN(KA)) 30,30,28
 28   IF(LIN(KC)) 58,58,29
 29   DF = 1.0
      ALP = D(I)
      INN = IN + 15
      IMM = IM + 15
 51   ASSIGN 56 TO IER
 50 RATIO = (ERDF*ALP)/(ERSS*DF)
    IF (55-LV) 61,61,62
 61 WRITE (6,41)(HEAD(J),J=1,10)
    LV = 0
 62 J = DF
    WRITE (6,64)HOLER(KA),HOLER(INN),HOLER(19),HOLER(KC),HOLER(IMM),J,
   1ALP,RATIO
    LV = LV+1
    GO TO IER,(54,56,60)
 58 DF = DF + 1.0
    ALP = ALP + D(I)
    IM = IM + 1
    IF (LEV(KC)-IM)60,59,14
 59 IM = 11
    IMM = 22
    INN = IN + 15
    ASSIGN 60 TO IER
    GO TO 50
 60 DF = 0.0
    ALP = 0.0
    GO TO 17
 30 DF = DF + 1.0
    ALP = ALP + D(I)
    IF (LIN(KC)) 39,39,52
 39 IM = IM + 1
    IF (LEV(KC) - IM) 40,40,14
 40 IM = 1
    IN = IN + 1
    IF (LEV(KA) - IN) 18,49,14
 49 IM = 11
    IN = 11
    INN = 22
    IMM = 22
    GO TO 51
 52 IN = IN + 1
    IF (LEV(KA)-IN) 53,53,14
 53 IN = 11
    INN = 22
    IMM = IM + 15

    ASSIGN 54 TO IER
    GO TO 50
 54 IM = IM + 1
    IF (LEV(KC) - IM) 18,18,57
 57 IF (IM-3) 55,55,18
 56 IM=IM+1
    IF (LEV(KC)-IM)17,17,20
 17 IN=IN+1
    IM=1
    IF(LEV(KA)-IN) 18,18,19
 19 IF (IN-3) 14,14,18
 20 IF (IM-3) 14,14,17
 14 CONTINUE
 18 CONTINUE
    IER=ERDF
    WRITE (6,44)IER,ERSS
    N = N-1
    WRITE (6,45)N,TSS
    DF = SQRT(ERSS/ERDF)
    WRITE (6,63)DF
    DO 22 I=1,M
    DO 22 J=I,M
 22 SSS(I)=SSS(I)+SS(J)
    WRITE (6,48)
    DO 23 I=1,M
 23 WRITE (6,46)I,SS(I)
    WRITE (6,42)
```

```
      DO 24 I=1,M
   24 WRITF (6,47)I,SSS(I)
      RETURN
      END
$IRFTC MEANS   LIST,REF,DGCK
      SUBROUTINE MEANS
      COMMON P(9,9,9),INDFX(15),KOG(15),LEV(15),LUMP(15),D(21000),LIP(15
     1),VECTOR(9),HEAD(10),HOLER(22),LEVEL,M,N,MF
   11 FORMAT (19H0      GRAND MEAN = E13.6)
   12 FORMAT (1X,A3,11X,5E20.6/15X,4E20.6)
   13 FORMAT (1H )
   19 FORMAT (1H1)
   21 FORMAT (1X,A3,A1,I1,A2,A3,4X,5E20.6/15X,4E20.6)
      VECTOR(1) = 0.0
      DO 8 K=1,N
    8 VECTOR(1) = VECTOR(1)+D(K)
      FLNUM = N
      VFCTOR(1) = VECTOR(1)/FLNUM
      WRITE (6,11)VECTOR(1)
      LEVEL = 1
      LV = 0
      DO 14 KF=MF,M
      WRITE (6,13)
      LV = LV+1
      LL = LEV(KF)
      DO 1 I=1,LL
    1 VECTOR(I) = 0.0
      DO 2 K=1,N
      J = KOG(KF)+1
      VECTOR(J) = VECTOR(J)+D(K)
      CALL COUNT(16)
    2 CONTINUF
      FLNUM = N/LL
      DO 3 J=1,LL
    3 VECTOR(J) = VECTOR(J)/FLNUM
      WRITE (6,12)HOLER(KF),(VFCTOR(J),J=1,LL)
      LV = LV+1+(LL/5)
      IF (KF-1) 14,14,5
    5 KG = KF+1
      IF (M-KG) 10,16,16
   16 DO 4 KH=KG,M
      IN = 0
      WRITE (6,13)
      LV = LV+1
      LF = LEV(KH)
      DO 4 KQ=1,LL
      IN = IN+1
      KS = KQ-1
      DO 6 I=1,LF
    6 VECTOR(I) = 0.0
      I = 0
      DO 17 K=1,N
      IF (KOG(KF)-KS) 7,15,7
   15 J = KOG(KH)+1
      VECTOR(J) = VECTOR(J)+D(K)
      I = I+1
    7 CALL COUNT(16)
   17 CONTINUE
      IF (LV-50) 18,20,20
   20 WRITE (6,19)
      LV = 0
   18 FLNUM = I/LF
      DO 9 I=1,LF
```

```
  9 VECTOR(I) = VECTOR(I)/FLNUM
    WRITE (6,21)HOLER(KF),HOLER(20),IN,HOLER(21),HOLER(KH),(VECTOR(I),
   1I=1,LF)
  4 LV = LV+1+(LF/5)
 14 CONTINUE
 10 RETURN
    END
```

Appendix D

GRAM-SCHMIDT CONSTRUCTION WITH NORMALIZATION

The following X matrix and identity matrix will be used for illustrating, step by step, their transformations into the P matrix and D matrix, respectively.

$$X = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{bmatrix}, \quad I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Retaining $x_{.1}$, the matrices remain the same, thus

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The second column is formed by the vector equation

$$q_{.2} = x_{.2} + c_{12} q_{.1}$$

where

$$c_{12} = \frac{-q_{.1} \cdot x_{.2}}{q_{.1} \cdot q_{.1}}$$

thus

$$q_{.1} \cdot q_{.1} = \sum_{i=1}^{3} \left(q_{i1}\right)^2 = 1^2 + 1^2 + 1^2 = 3$$

$$q_{.1} \cdot x_{.2} = \sum_{i=1}^{3} q_{i1} x_{i2} = 1 \times 1 + 1 \times 2 + 1 \times 3 = 6$$

$$c_{12} = -\frac{6}{3} = -2$$

and the matrices become

$$\begin{bmatrix} 1 & 1 - 2 \times 1 & 1 \\ 1 & 2 - 2 \times 1 & 4 \\ 1 & 3 - 2 \times 1 & 9 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 - 2 \times 1 & 0 \\ 0 & 1 - 2 \times 0 & 0 \\ 0 & 0 - 2 \times 0 & 1 \end{bmatrix}$$

which, upon computing the second columns, become

$$\begin{bmatrix} 1 & -1 & 1 \\ 1 & 0 & 4 \\ 1 & 1 & 9 \end{bmatrix} \quad \begin{bmatrix} 1 & -2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The third column is formed by the vector equation

$$q_{.3} = x_{.3} + c_{13} q_{.1} + c_{23} q_{.2}$$

where

$$c_{13} = \frac{-q_{.1} \cdot x_{.3}}{q_{.1} \cdot q_{.1}}$$

and

$$c_{23} = \frac{-q_{.2} \cdot x_{.3}}{q_{.2} \cdot q_{.2}}$$

thus

$$q_{.2} \cdot q_{.2} = \sum_{i=1}^{3} \left(q_{i2}\right)^2 = (-1)^2 + 0^2 + 1^2 = 2$$

$$q_{.1} \cdot x_{.3} = \sum_{i=1}^{3} q_{i1} x_{i3} = 1 \times 1 + 1 \times 4 + 1 \times 9 = 14$$

$$q_{.2} \cdot x_{.3} = \sum_{i=1}^{3} q_{i2} x_{i3} = - 1 \times 1 + 0 \times 4 + 1 \times 9 = 8$$

$$c_{13} = \frac{-14}{3}$$

$$c_{23} = \frac{-8}{2} = -4$$

and the matrices become

$$\begin{bmatrix} 1 & -1 & 1 - \frac{14}{3} \times 1 - 4(-1) \\ 1 & 0 & 4 - \frac{14}{3} \times 1 - 4 \times 0 \\ 1 & 1 & 9 - \frac{14}{3} \times 1 - 4 \times 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & -2 & 0 - \frac{14}{3} \times 1 - 4(-2) \\ 0 & 1 & 0 - \frac{14}{3} \times 0 - 4 \times 1 \\ 0 & 0 & 1 - \frac{14}{3} \times 0 - 4 \times 0 \end{bmatrix}$$

which, upon computing the third columns, become

$$\begin{bmatrix} 1 & -1 & \frac{1}{3} \\ 1 & 0 & -\frac{2}{3} \\ 1 & 1 & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 1 & -2 & \frac{10}{3} \\ 0 & 1 & -4 \\ 0 & 0 & 1 \end{bmatrix}$$

This completes the Gram-Schmidt construction.

The left matrix will be normalized to obtain the *P* matrix, and the right matrix will become the *D* matrix. The first column is formed by the equation

$$p_{i1} = \frac{q_{i1}}{r_{11}}$$

where

$$r_{11} = |q_{.1}|^{-1} = \left(\sqrt{\sum_{i=1}^{3}\left(q_{i1}\right)^{2}}\right)^{-1}$$

thus

$$r_{11} = \left(\sqrt{1^2 + 1^2 + 1^2}\right)^{-1} = \sqrt{\frac{1}{3}}$$

and the matrices become:

$$\begin{bmatrix} \sqrt{\frac{1}{3}} & 1 & \frac{1}{3} \\ \sqrt{\frac{1}{3}} & 0 & \frac{2}{3} \\ \sqrt{\frac{1}{3}} & 1 & \frac{1}{3} \end{bmatrix} \begin{bmatrix} \sqrt{\frac{1}{3}} & 2 & \frac{10}{3} \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{bmatrix}$$

The second column is formed by the equation

$$p_{i2} = \frac{q_{i2}}{r_{22}}$$

where

$$r_{22} = |q_{.2}|^{-1} = \left(\sqrt{\sum_{i=1}^{3} q_{i2}}\right)^{-1}$$

thus

$$r_{22} = \left(\sqrt{(-1)^2 + 0^2 + 1^2}\right)^{-1} = \sqrt{\frac{1}{2}}$$

and the matrices become

$$\begin{bmatrix} \sqrt{\frac{1}{3}} & -\sqrt{\frac{1}{2}} & \frac{1}{3} \\ \sqrt{\frac{1}{3}} & 0 & -\frac{2}{3} \\ \sqrt{\frac{1}{3}} & \sqrt{\frac{1}{2}} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} \sqrt{\frac{1}{3}} & -2\sqrt{\frac{1}{2}} & \frac{10}{3} \\ 0 & \sqrt{\frac{1}{2}} & 4 \\ 0 & 0 & 1 \end{bmatrix}$$

The third column is formed by the equation

$$p_{i3} = \frac{q_{i3}}{r_{33}}$$

where

$$r_{33} = |q_{.3}|^{-1} = \left(\sqrt{\sum_{i=1}^{3} q_{i3}}\right)^{-1}$$

thus

$$r_{33} = \left(\sqrt{\left(\frac{1}{3}\right)^2 + \left(-\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2}\right)^{-1} = 3\sqrt{\frac{1}{6}}$$

and the matrices become

$$\begin{bmatrix} \sqrt{\frac{1}{3}} & -\sqrt{\frac{1}{2}} & \sqrt{\frac{1}{6}} \\ \sqrt{\frac{1}{3}} & 0 & -2\sqrt{\frac{1}{6}} \\ \sqrt{\frac{1}{3}} & \sqrt{\frac{1}{2}} & \sqrt{\frac{1}{6}} \end{bmatrix} \begin{bmatrix} \sqrt{\frac{1}{3}} & -2\sqrt{\frac{1}{2}} & 10\sqrt{\frac{1}{6}} \\ 0 & \sqrt{\frac{1}{2}} & -12\sqrt{\frac{1}{6}} \\ 0 & 0 & 3\sqrt{\frac{1}{6}} \end{bmatrix}$$

This completes the transformation of the X matrix and of the identity matrix into the P matrix and into the D matrix, respectively. The P matrix is

$$\begin{bmatrix} \sqrt{\frac{1}{3}} & -\sqrt{\frac{1}{2}} & \sqrt{\frac{1}{6}} \\ \sqrt{\frac{1}{3}} & 0 & -2\sqrt{\frac{1}{6}} \\ \sqrt{\frac{1}{3}} & \sqrt{\frac{1}{2}} & \sqrt{\frac{1}{6}} \end{bmatrix}$$

and the D matrix is

$$\begin{bmatrix} \sqrt{\frac{1}{3}} & -2\sqrt{\frac{1}{2}} & 10\sqrt{\frac{1}{6}} \\ 0 & \sqrt{\frac{1}{2}} & -12\sqrt{\frac{1}{6}} \\ 0 & 0 & 3\sqrt{\frac{1}{6}} \end{bmatrix}$$

Appendix E

TENSOR PRODUCT

The following equation will be used for illustrating the product of three tensors

$$\underline{A}_s^i \, \underline{B}_t^j \, \underline{C}_{ij} = \underline{D}_{st}$$

where

$$1 \leq i \leq 4$$
$$1 \leq j \leq 3$$
$$1 \leq s \leq 4$$
$$1 \leq t \leq 3$$

The elements of $\underline{A}_s^i$ may be arranged as

$$\underline{A}_s^i \equiv \begin{bmatrix} a_1^1 & a_2^1 & a_3^1 & a_4^1 \\ a_1^2 & a_2^2 & a_3^2 & a_4^2 \\ a_1^3 & a_2^3 & a_3^3 & a_4^3 \\ a_1^4 & a_2^4 & a_3^4 & a_4^4 \end{bmatrix}$$

The elements of $\underline{B}_t^j$ may be arranged as

$$\underline{B}_t^j \equiv \begin{bmatrix} b_1^1 & b_2^1 & b_3^1 \\ b_1^2 & b_2^2 & b_3^2 \\ b_1^3 & b_2^3 & b_3^3 \end{bmatrix}$$

The elements of $\underline{C}_{ij}$ may be arranged as

$$\underline{C}_{ij} \equiv \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \\ c_{41} & c_{42} & c_{43} \end{bmatrix}$$

The elements of $\underline{D}_{st}$ may be arranged as

$$\underline{D}_{st} \equiv \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{22} & d_{33} \\ d_{41} & d_{42} & d_{43} \end{bmatrix}$$

The product $\underline{B}_t^j\ \underline{C}_{ij}$ may be formed first and set equal to a tensor $\underline{E}_{ti}$. Note that the j is summed out.

$$\underline{B}_t^j\ \underline{C}_{ij} = \underline{E}_{ti} \equiv \begin{bmatrix} e_{11} & e_{21} & e_{31} \\ e_{12} & e_{22} & e_{32} \\ e_{13} & e_{23} & e_{33} \\ e_{14} & e_{24} & e_{34} \end{bmatrix}$$

where

$$\begin{aligned}
e_{11} &= b_1^1\, c_{11} + b_1^2\, c_{12} + b_1^3\, c_{13} \\
e_{21} &= b_2^1\, c_{11} + b_2^2\, c_{12} + b_2^3\, c_{13} \\
e_{31} &= b_3^1\, c_{11} + b_3^2\, c_{12} + b_3^3\, c_{13} \\
e_{12} &= b_1^1\, c_{21} + b_1^2\, c_{22} + b_1^3\, c_{23} \\
e_{22} &= b_2^1\, c_{21} + b_2^2\, c_{22} + b_2^3\, c_{23} \\
e_{32} &= b_3^1\, c_{21} + b_3^2\, c_{22} + b_3^3\, c_{23} \\
e_{13} &= b_1^1\, c_{31} + b_1^2\, c_{32} + b_1^3\, c_{33} \\
e_{23} &= b_2^1\, c_{31} + b_2^2\, c_{32} + b_2^3\, c_{33} \\
e_{33} &= b_3^1\, c_{31} + b_3^2\, c_{32} + b_3^3\, c_{33} \\
e_{14} &= b_1^1\, c_{41} + b_1^2\, c_{42} + b_1^3\, c_{43} \\
e_{24} &= b_2^1\, c_{41} + b_2^2\, c_{42} + b_2^3\, c_{43} \\
e_{34} &= b_3^1\, c_{41} + b_3^2\, c_{42} + b_3^3\, c_{43}
\end{aligned}$$

The tensor $\underline{D}_{st}$ may be formed from the product $\underline{A}_s^i \underline{E}_{ti}$ where

$$d_{11} = a_1^1 e_{11} + a_1^2 e_{12} + a_1^3 e_{13} + a_1^4 e_{14}$$

$$d_{12} = a_1^1 e_{21} + a_1^2 e_{22} + a_1^3 e_{23} + a_1^4 e_{24}$$

$$d_{13} = a_1^1 e_{31} + a_1^2 e_{32} + a_1^3 e_{33} + a_1^4 e_{34}$$

$$d_{21} = a_2^1 e_{11} + a_2^2 e_{12} + a_2^3 e_{13} + a_2^4 e_{14}$$

$$d_{22} = a_2^1 e_{21} + a_2^2 e_{22} + a_2^3 e_{23} + a_2^4 e_{24}$$

$$d_{23} = a_2^1 e_{31} + a_2^2 e_{32} + a_2^3 e_{33} + a_2^4 e_{34}$$

$$d_{31} = a_3^1 e_{11} + a_3^2 e_{12} + a_3^3 e_{13} + a_3^4 e_{14}$$

$$d_{32} = a_3^1 e_{21} + a_3^2 e_{22} + a_3^3 e_{23} + a_3^4 e_{24}$$

$$d_{33} = a_3^1 e_{31} + a_3^2 e_{32} + a_3^3 e_{33} + a_3^4 e_{34}$$

$$d_{41} = a_4^1 e_{11} + a_4^2 e_{12} + a_4^3 e_{13} + a_4^4 e_{14}$$

$$d_{42} = a_4^1 e_{21} + a_4^2 e_{22} + a_4^3 e_{23} + a_4^4 e_{24}$$

$$d_{43} = a_4^1 e_{31} + a_4^2 e_{32} + a_4^3 e_{33} + a_4^4 e_{34}$$

Instead of having formed the tensor product $\underline{B}_t^j \underline{C}_{ij}$ first, the product $\underline{A}_s^i \underline{C}_{ij}$ could have been formed and then the tensor $\underline{D}_{st}$.

A third way of arriving at the tensor $\underline{D}_{st}$ is to perform first the tensor product $\underline{A}_s^i \underline{B}_t^j$. This tensor product has, for the example, 144 elements, some of which follow.

$$\underline{A}^i_s \underline{B}^j_t \equiv \begin{bmatrix}
a^1_1 b^1_1 & a^1_2 b^1_1 & a^1_3 b^1_1 & a^1_4 b^1_1 \\
a^1_1 b^2_1 & a^1_2 b^2_1 & a^1_3 b^2_1 & a^1_4 b^2_1 \\
a^1_1 b^3_1 & a^1_2 b^3_1 & a^1_3 b^3_1 & a^1_4 b^3_1 \\
a^1_1 b^1_2 & a^1_2 b^1_2 & a^1_3 b^1_2 & a^1_4 b^1_2 \\
a^1_1 b^2_2 & a^1_2 b^2_2 & a^1_3 b^2_2 & a^1_4 b^2_2 \\
a^1_1 b^3_2 & a^1_2 b^3_2 & a^1_3 b^3_2 & a^1_4 b^3_2 \\
a^1_1 b^1_3 & a^1_2 b^1_3 & a^1_3 b^1_3 & a^1_4 b^1_3 \\
a^1_1 b^2_3 & a^1_2 b^2_3 & a^1_3 b^2_3 & a^1_4 b^2_3 \\
a^1_1 b^3_3 & a^1_2 b^3_3 & a^1_3 b^3_3 & a^1_4 b^3_3 \\
a^2_1 b^1_1 & a^2_2 b^1_1 & a^2_3 b^1_1 & a^2_4 b^1_1 \\
a^2_1 b^2_1 & a^2_2 b^2_1 & a^2_3 b^2_1 & a^2_4 b^2_1 \\
\cdots & \cdots & \cdots & \cdots \\
a^2_1 b^3_3 & a^2_2 b^3_3 & a^2_3 b^3_3 & a^2_4 b^3_3 \\
a^3_1 b^1_1 & a^3_2 b^1_1 & a^3_3 b^1_1 & a^3_4 b^1_1 \\
\cdots & \cdots & \cdots & \cdots \\
a^3_1 b^3_3 & a^3_2 b^3_3 & a^3_3 b^3_3 & a^3_4 b^3_3 \\
a^4_1 b^1_1 & a^4_2 b^1_1 & a^4_3 b^1_1 & a^4_4 b^1_1 \\
\cdots & \cdots & \cdots & \cdots \\
a^4_1 b^3_3 & a^4_2 b^3_3 & a^4_3 b^3_3 & a^4_4 b^3_3
\end{bmatrix}$$

Multiplying the above tensor by $\underline{C}_{ij}$, the tensor $\underline{D}_{st}$ is obtained, the elements of which are

$$\begin{aligned}
d_{11} = {} & a^1_1 b^1_1 c_{11} + a^1_1 b^2_1 c_{12} + a^1_1 b^3_1 c_{13} + \\
& a^2_1 b^1_1 c_{21} + a^2_1 b^2_1 c_{22} + a^2_1 b^3_1 c_{23} + \\
& a^3_1 b^1_1 c_{31} + a^3_1 b^2_1 c_{32} + a^3_1 b^3_1 c_{33} + \\
& a^4_1 b^1_1 c_{41} + a^4_1 b^2_1 c_{42} + a^4_1 b^3_1 c_{43}
\end{aligned}$$

$$\begin{aligned} d_{12} = {} & a_1^1 b_2^1 c_{11} + a_1^1 b_2^2 c_{12} + a_1^1 b_2^3 c_{13} + \\ & a_1^2 b_2^1 c_{21} + a_1^2 b_2^2 c_{22} + a_1^2 b_2^3 c_{23} + \\ & a_1^3 b_2^1 c_{31} + a_1^3 b_2^2 c_{32} + a_1^3 b_2^3 c_{33} + \\ & a_1^4 b_2^1 c_{41} + a_1^4 b_2^2 c_{42} + a_1^4 b_2^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{13} = {} & a_1^1 b_3^1 c_{11} + a_1^1 b_3^2 c_{12} + a_1^1 b_3^3 c_{13} + \\ & a_1^2 b_3^1 c_{21} + a_1^2 b_3^2 c_{22} + a_1^2 b_3^3 c_{23} + \\ & a_1^3 b_3^1 c_{31} + a_1^3 b_3^2 c_{32} + a_1^3 b_3^3 c_{33} + \\ & a_1^4 b_3^1 c_{41} + a_1^4 b_3^2 c_{42} + a_1^4 b_3^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{21} = {} & a_2^1 b_1^1 c_{11} + a_2^1 \,{}_1^2 c_{12} + a_2^1 b_1^3 c_{13} + \\ & a_2^2 b_1^1 c_{21} + a_2^2 b_1^2 c_{22} + a_2^2 b_1^3 c_{23} + \\ & a_2^3 b_1^1 c_{31} + a_2^3 b_1^2 c_{32} + a_2^3 b_1^3 c_{33} + \\ & a_2^4 b_1^1 c_{41} + a_2^4 b_1^2 c_{42} + a_2^4 b_1^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{22} = {} & a_2^1 b_2^1 c_{11} + a_2^1 b_2^2 c_{12} + a_2^1 b_2^3 c_{13} + \\ & a_2^2 b_2^1 c_{21} + a_2^2 b_2^2 c_{22} + a_2^2 b_2^3 c_{23} + \\ & a_2^3 b_2^1 c_{31} + a_2^3 b_2^2 c_{32} + a_2^3 b_2^3 c_{33} + \\ & a_2^4 b_2^1 c_{41} + a_2^4 b_2^2 c_{42} + a_2^4 b_2^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{23} = {} & a_2^1 b_3^1 c_{11} + a_2^1 b_3^2 c_{12} + a_2^1 b_3^3 c_{13} + \\ & a_2^2 b_3^1 c_{21} + a_2^2 b_3^2 c_{22} + a_2^2 b_3^3 c_{2} + \\ & a_2^3 b_3^1 c_{31} + a_2^3 b_3^2 c_{32} + a_2^3 b_3^3 c_{33} + \\ & a_2^4 b_3^1 c_{41} + a_2^4 b_3^2 c_{42} + a_2^4 b_3^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{31} = {} & a_3^1 b_1^1 c_{11} + a_3^1 b_1^2 c_{12} + a_3^1 b_1^3 c_{13} + \\ & a_3^2 b_1^1 c_{21} + a_3^2 b_1^2 c_{22} + a_3^2 b_1^3 c_{23} + \\ & a_3^3 b_1^1 c_{31} + a_3^3 b_1^2 c_{32} + a_3^3 b_1^3 c_{33} + \\ & a_3^4 b_1^1 c_{41} + a_3^4 b_1^2 c_{42} + a_3^4 b_1^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{32} = {} & a_3^1 b_2^1 c_{11} + a_3^1 b_2^2 c_{12} + a_3^1 b_2^3 c_{13} + \\ & a_3^2 b_2^1 c_{21} + a_3^2 b_2^2 c_{22} + a_3^2 b_2^3 c_{23} + \\ & a_3^3 b_2^1 c_{31} + a_3^3 b_2^2 c_{32} + a_3^3 b_2^3 c_{33} + \\ & a_3^4 b_2^1 c_{41} + a_3^4 b_2^2 c_{42} + a_3^4 b_2^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{33} = {} & a_3^1 b_3^1 c_{11} + a_3^1 b_3^2 c_{12} + a_3^1 b_3^3 c_{13} + \\ & a_3^2 b_3^1 c_{21} + a_3^2 b_3^2 c_{22} + a_3^2 b_3^3 c_{23} + \\ & a_3^3 b_3^1 c_{31} + a_3^3 b_3^2 c_{32} + a_3^3 b_3^3 c_{33} + \\ & a_3^4 b_3^1 c_{41} + a_3^4 b_3^2 c_{42} + a_3^4 b_3^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{41} = {} & a_4^1 b_1^1 c_{11} + a_4^1 b_1^2 c_{12} + a_4^1 b_1^3 c_{13} + \\ & a_4^2 b_1^1 c_{21} + a_4^2 b_1^2 c_{22} + a_4^2 b_1^3 c_{23} + \\ & a_4^3 b_1^1 c_{31} + a_4^3 b_1^2 c_{32} + a_4^3 b_1^3 c_{33} + \\ & a_4^4 b_1^1 c_{41} + a_4^4 b_1^2 c_{42} + a_4^4 b_1^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{42} = {} & a_4^1 b_2^1 c_{11} + a_4^1 b_2^2 c_{12} + a_4^1 b_2^3 c_{13} + \\ & a_4^2 b_2^1 c_{21} + a_4^2 b_2^2 c_{22} + a_4^2 b_2^3 c_{23} + \\ & a_4^3 b_2^1 c_{31} + a_4^3 b_2^2 c_{32} + a_4^3 b_2^3 c_{33} + \\ & a_4^4 b_2^1 c_{41} + a_4^4 b_2^2 c_{42} + a_4^4 b_2^3 c_{43} \end{aligned}$$

$$\begin{aligned} d_{43} = \; & a^1_4\, b^1_3\, c_{11} + a^1_4\, b^2_3\, c_{12} + a^1_4\, b^3_3\, c_{13} + \\ & a^2_4\, b^1_3\, c_{21} + a^2_4\, b^2_3\, c_{22} + a^2_4\, b^3_3\, c_{23} + \\ & a^3_4\, b^1_3\, c_{31} + a^3_4\, b^2_3\, c_{32} + a^3_4\, b^3_3\, c_{33} + \\ & a^4_4\, b^1_3\, c_{41} + a^4_4\, b^2_3\, c_{42} + a^4_4\, b^3_3\, c_{43} \end{aligned}$$

A numerical example is given to illustrate the above operations of tensor product. The elements of each tensor are arranged in matrix form. The numerical values of the elements are given for the $\underline{A}^i_s$, $\underline{B}^j_t$, and $\underline{C}_{ij}$ *tensors.*

$$\underline{A}^i_s = \begin{bmatrix} \frac{1}{2} & \frac{-3}{2\sqrt{5}} & \frac{1}{2} & \frac{-1}{2\sqrt{5}} \\ \frac{1}{2} & \frac{-1}{2\sqrt{5}} & \frac{-1}{2} & \frac{3}{2\sqrt{5}} \\ \frac{1}{2} & \frac{1}{2\sqrt{5}} & -\frac{1}{2} & \frac{-3}{2\sqrt{5}} \\ \frac{1}{2} & \frac{3}{2\sqrt{5}} & \frac{1}{2} & \frac{1}{2\sqrt{5}} \end{bmatrix}$$

$$\underline{B}^j_t = \begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & 0 & \frac{-2}{\sqrt{6}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \end{bmatrix}$$

$$\underline{C}_{ij} = \begin{bmatrix} 1 & 2 & 4 \\ 3 & 3 & 4 \\ 4 & 5 & 7 \\ 6 & 7 & 8 \end{bmatrix}$$

As has been outlined above, there are three different ways of arriving at the $\underline{D}_{st}$ tensor. The first method will be given, where the $\underline{E}_{ti}$ tensor is formed first, and then the $\underline{D}_{st}$ tensor is computed.

The $\underline{E}_{ti}$ *tensor is*

$$\begin{bmatrix} \frac{7}{\sqrt{3}} & \frac{3}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ \frac{10}{\sqrt{3}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ \frac{16}{\sqrt{3}} & \frac{3}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ \frac{21}{\sqrt{3}} & \frac{2}{\sqrt{2}} & 0 \end{bmatrix}$$

The formulas for the elements of this tensor were given previously.

The D_{st} tensor is

$$\begin{bmatrix} \frac{27}{\sqrt{3}} & \frac{9}{2\sqrt{2}} & \frac{3}{2\sqrt{6}} \\ \frac{24}{\sqrt{15}} & \frac{-1}{2\sqrt{10}} & \frac{-3}{2\sqrt{30}} \\ \frac{1}{\sqrt{3}} & \frac{1}{2\sqrt{2}} & \frac{-1}{2\sqrt{6}} \\ \frac{-2}{\sqrt{15}} & \frac{-7}{2\sqrt{10}} & \frac{-1}{2\sqrt{30}} \end{bmatrix}$$

The formulas for the elements of this tensor were given previously.

REFERENCES

1. Cruise, D. R., and S. R. Miller. "NOVAN - A Variance Analysis Program for Factorial Experiments," presented at the 14th national meeting of the Association for Computing Machinery, Cambridge, Mass., September 1959. (Available from ACM headquarters, New York, N.Y.)

2. U.S. Naval Ordnance Test Station. Development of Data Smoothing Techniques for Digital Computers Using Linear Methods, by D. R. Cruise. China Lake, Calif., NOTS, 30 April 1962. (NAVWEPS Report 7818, NOTS TP 2826.)

3. Stoll, R. R. Linear Algebra and Matrix Theory. New York, McGraw-Hill, 1952. Chap. 8, Sect. 8.7.

4. Anderson, R. L., and T. A. Bancroft. Statistical Theory in Research. New York, McGraw-Hill, 1952. Chap. 16.

5. Scheffé, Henry. The Analysis of Variance. New York, Wiley, 1959. Chap. 16.

6. Solkolnikoff, I. S. Tensor Analysis. New York, Wiley, 1958. 385 pp.

7. Davies, O. L. The Design and Analysis of Industrial Experiments. London, Oliver and Boyd, 1960. Chap. 8.

8. Stewart, B. M. Theory of Numbers. New York, MacMillan, 1952. Chap. 4.

9. Villars, Donald S. Statistical Design and Analysis of Experiments for Development Research. Dubuque, Iowa, Wm. C. Brown Co., 1951. Chap. 12, Sect. 2.

10. Devalon, C. C., Jr. "NO VALU - A Program to Compute Missing Values in a Designed Variance Analysis," presented at the 14th national meeting of the Association for Computing Machinery, Cambridge, Mass., September 1959. (Available from ACM headquarters, New York, N.Y.)

INITIAL DISTRIBUTION

28 Naval Air Systems Command

AIR-03A (1)
AIR-503 (1)
AIR-5035 (1)
AIR-604 (2)
CS (1)
DMP-33 (1)
FWAM (1)
FWTP (1)
P-1 (1)
PWS-3 (1)
R-12 (1)
RAAV-5 (1)
RAPP (1)
RM-1 (1)
RM-3 (1)
RMMP (1)
RMMP-2 (1)
RMMP-24 (1)
RMMP-33 (1)
RMMP-4 (1)
RMMP-42 (1)
RRMA-221 (1)
RRRE (1)
RRRE-5 (1)
RT (1)
RUME-11 (1)
RUSE-2 (1)

5 Chief of Naval Operations
Deputy Chief for Air (1)
Operations Evaluation Group (1)

2 Office of Naval Research
Code 104 (1)
Code 463 (1)

1 Assistant Secretary of the Navy (Installations and Logistics)
1 David W Taylor Model Basin (Aerodynamics Laboratory, Code 046)
1 Fleet Anti-Air Warfare Training Center, San Diego (Guided Missile Section)
1 Naval Air Development Center, Johnsville (Technical Library)
1 Naval Air Engineering Center, Philadelphia
1 Naval Air Force, Atlantic Fleet
1 Naval Air Force, Pacific Fleet
1 Naval Air Test Center, Patuxent River (Aeronautical Publications Library)
1 Naval Avionics Facility, Indianapolis (Technical Library)
1 Naval Ordnance Laboratory, White Oak (Technical Library)
2 Naval Propellant Plant, Indian Head
1 Naval Research Laboratory (Code 2021)
2 Naval Underwater Weapons Research and Engineering Station, Newport
1 Naval Weapons Laboratory, Dahlgren (Technical Library)
2 Naval Weapons Services Office(Code DM)
1 Operational Test and Evaluation Force
1 Bureau of Naval Weapons Fleet Readiness Representative, Pacific
1 Bureau of Naval Weapons Representative, Azusa
1 Navy Liaison Officer, Tactical Air Command, Langley Air Force Base
1 Aberdeen Proving Ground (Development and Proof Services)
1 Army Ballistics Research Laboratories, Aberdeen Proving Ground
7 Army Missile Command, Redstone Arsenal
ABMA (1)
ORDDW-IDE (1)
ORDXR-ODA (1)
Redstone Scientific Information Center (4)

2 Frankford Arsenal
ORDBA-1421 (1)
Pitman-Dunn Laboratory (1)

1 Harry Diamond Laboratories (Technical Library)
1 Picatinny Arsenal (Technical Library)
1 Rock Island Arsenal
3 White Sands Missile Range (ORDBS-Technical Library)
1 Headquarters, U S Air Force (AFRSTD)
1 Tactical Air Command, Langley Air Force Base (TPL-RQD-M)
1 Ogden Air Materiel Area, Hill Air Force Base (OOYD)

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author)
U.S. Naval Ordnance Test Station
China Lake, California 93555

2a REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b GROUP

3 REPORT TITLE
COMPUTING SCHEMES FOR FACTORIAL AND REGRESSION ANALYSIS AND DEVELOPMENT OF THE UNDERLYING THEORY USING LINEAR METHODS

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)
Summary report for FY 1957 through FY 1965

5 AUTHOR(S) (Last name, first name, initial)
Miller, S. R.
Cruise, D. R.

6 REPORT DATE
November 1966

7a TOTAL NO. OF PAGES
68

7b NO. OF REFS
10

8a CONTRACT OR GRANT NO.
RMMP-22-028/216-1/F009-06-01

b PROJECT NO.

c.

d.

9a ORIGINATOR'S REPORT NUMBER(S)
NOTS TP 3974

9b OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
NAVWEPS Report 9001

10 AVAILABILITY/LIMITATION NOTICES
Distribution of this document is unlimited.

11 SUPPLEMENTARY NOTES
Appendixes A, B, C, D, and E

12 SPONSORING MILITARY ACTIVITY
Naval Ordnance Systems Command
Naval Material Command
Washington, D. C., 20360

13 ABSTRACT

The basic concepts of regression and factorial analysis are reviewed and the theory developed, using the notation of linear algebra. The development includes the Gram-Schmidt construction, the linear method of least squares, and the use of orthogonal polynomials. Computing schemes applicable to digital computers are developed, and the basic coding is presented in the FORTRAN language. Fractional replication and the missing value problem are discussed.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Curvilinear Regression						
Statistical Design						
Factorial Analysis						
Factorial Designs						
Fractional Replication						
Gram-Schmidt Construction						
Lattice Designs						
Least Squares, Linear Method of						
Missing Value						
Multilinear Regression						
Orthogonal Polynomials						
Regression Analysis						
Tensor						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Date" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through

____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through

____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through

____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13 ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14 KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.