MEMORANDUM
RM-4879-PR
JULY 1967

MULTIVARIATE LOGARITHMIC AND EXPONENTIAL REGRESSION MODELS

C. A. Graver and H. E. Boren, Jr.

PREPARED FOR:
UNITED STATES AIR FORCE PROJECT RAND

The RAND Corporation
SANTA MONICA • CALIFORNIA

MEMORANDUM
RM-4879-PR
JULY 1967

MULTIVARIATE LOGARITHMIC AND EXPONENTIAL REGRESSION MODELS

C. A. Graver and H. E. Boren, Jr.

This research is supported by the United States Air Force under Project RAND—Contract No. F44620-67-C-0045—monitored by the Directorate of Operational Requirements and Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF. Views or conclusions contained in this Memorandum should not be interpreted as representing the official opinion or policy of the United States Air Force.

The RAND Corporation
1700 MAIN ST. • SANTA MONICA • CALIFORNIA • 90406

Published by The RAND Corporation

PREFACE

This Memorandum is primarily the result of research being conducted in the Cost Analysis Department and the Mathematics Department of The RAND Corporation on the general use of statistics in the field of cost analysis. Cost analysis work involves many types of estimating relationships that are derived from historical data and that functionally relate resources to one or more system parameters. However, a relationship is not very meaningful unless one can indicate how well it will predict future resource requirements. To accomplish this, appropriate statistical measures must be determined, and the statistical assumptions implicit in the derivation of the estimating relationship must be clearly understood and satisfied.

The authors wish to acknowledge the generous help of Dr. Albert Madansky (now of Market Planning Corporation, New York) whose overall guidance and careful review, comments, and additions to the manuscript contributed greatly to this study. Dr. Madansky also assisted in the development of the computer program by devising special techniques that eliminated problems inherent in some of the matrix operations.

Although the mathematical models contained in this Memorandum were completed in January 1965, this publication was delayed because of several programming problems. The most serious of these involved a RAND auxiliary subroutine that generates eigenvalues and eigenvectors. Some of the matrices used in one of the models discussed in this Memorandum were too degenerate for that subroutine to handle. As a result, Mr. John Derr, of the RAND Computer Sciences Department, modified the subroutine so that it would generate satisfactory eigenvalues and eigenvectors

in the computer program. Mr. Derr's efforts contributed greatly to the success of the development of the computer program and are deeply appreciated by the authors.

Thanks are also due to Mr. Charles H. Bush of the RAND Computer Sciences Department for his development of the variable format procedure used in the computer program.

SUMMARY

This Memorandum presents a statistical study of a regression function of the form

$$Y = e^{\mu_0} X_1^{\mu_1} X_2^{\mu_2} \ldots X_p^{\mu_p}.$$

The study is concerned primarily with (1) how to pick the "best" values for the unknown parameters (μ_0, μ_1, μ_2, ..., μ_p) given a set of historical data (Y_i, X_{1i}, X_{2i}, ..., X_{pi}, i = 1, 2, ..., n), and (2) how to describe the predictive capability of the choice in (1). In order to study problems (1) and (2) above, the error assumption on Y must be described explicitly.

For this study, two basic types of error assumptions are considered--multiplicative and additive. In the multiplicative model, the error term is assumed to have a log-normal distribution and acts as a multiplier to the hypothetical regression function. This assumption is shown to lead to the usual treatment of problem (1)--taking natural logarithms of the regression relationship and then using linear least-squares estimators for estimates of the parameters.

Two forms of the multiplicative model are treated. In one form, the hypothetical regression function is assumed to be equal to the median of Y, whereas in the other form, the function is assumed to be equal to the expected value (mean) of Y. The study shows, however, that the usual logarithmic treatment actually represents the median form. It is also shown that the choice of the above forms of the multiplicative model affects only the estimate of μ_0 and its distribution, and does not affect the estimate of the other parameters (μ_1, μ_2, ..., μ_p), their distributions, or the prediction interval of Y.

For the additive model, the error is assumed to be normally distributed and additive to the hypothetical regression function. This assumption leads to the problem of least-squares estimation of a nonlinear form. The solution of the estimators for the additive model is not exact and may not be unique. It is shown, however, that for the type of regression function under consideration in this study, an absolute minimum of $\Sigma_{i=1}^{n} (Y_i - f_i)^2$ can be found for those values of $[\mu_0, \mu_1, \ldots, \mu_p]$ that satisfy

$$2f_i - Y_i > 0,$$

where

$$f_i = e^{\mu_0} X_{1i}^{\mu_1} X_{2i}^{\mu_2} \cdots X_{pi}^{\mu_p},$$

and Y_i = observed value at each point for $i = 1, 2, \ldots, n$.

For both models, methods of obtaining the estimators, distributions of the estimators, and prediction intervals for the dependent variable, Y, are given. Also, methods of comparing the two models are discussed.

A computer program was developed to calculate the estimators, distributions of the estimators, and prediction intervals, as described in this Memorandum. A discussion of the computer program, presented in Section V, deals with program restrictions, sequence of operations, input procedures, and program outputs.

One of the significant findings of this study is that care must always be taken in comparing different types of estimating relationships. Since the derivations of these estimating relationships may be based on different principles or error assumptions, the same statistical measures may not be applicable in all cases, and, if used, may lead to misleading conclusions.

CONTENTS

I. INTRODUCTION

Given a set of historical data that includes observations on a dependent variable and independent variables of interest, and having already decided on the form of the relationship involved, the analyst is usually confronted with the following closely related problems:

1. Picking the "best" values for the unknown parameters in the chosen relationship.
2. Describing the predictive capability of the choice in 1.

In this Memorandum these problems are analyzed for the following relationship:

$$Y \approx e^{\mu_0} X_1^{\mu_1} X_2^{\mu_2} \cdots X_p^{\mu_p}.$$

The quantity on the right of the symbol $\approx$ is called the "hypothetical regression function." It consists of independent variables (X_1, X_2, ..., X_p), the values of which are given to the analyst and are assumed to be positive and have no associated error, and parameters (μ_0, μ_1, ..., μ_p), which are unknown. The symbol Y is the dependent variable of interest, and it is assumed that the symbol $\approx$ would be replaced by an equality sign if there were no error associated with the observation of Y.

The assumption of error associated with the observation of Y is the reason for problems 1 and 2. For if this assumption were dropped, then all that the analyst would need is $p + 1$ data sets (observations) of the form $(Y_i, X_{1i}, X_{2i}, \ldots, X_{pi})$, $i = 1, \ldots, p + 1$, to determine the $p + 1$ parameters $\mu_0, \mu_1, \ldots, \mu_p$ exactly. Then, of course, the analyst could calculate (predict with no error) the Y value associated

with any values of $(X_1, X_2, \ldots, X_p)$. The "best" choice of the parameters would obviously have been made and the predictive capability would be perfect. However, with the error assumption on Y included in the model, the analyst must not only determine the "best" estimates of the unknown parameters (problem 1), but even if he were given the true values of these parameters, his predicted value of Y would in general still be different from the observed value of Y, and hence problem 2 would still exist. It is therefore apparent that the error assumption on Y is of great importance.

In order to study problems 1 and 2, the analyst must explicitly describe the error assumption on Y. Two forms of the error assumption will be discussed in this study. In the section entitled "The Multiplicative Model," the error is assumed to have a log-normal distribution and acts as a multiplier to the hypothetical regression function. This assumption will lead to the usual treatment of problem 1, i.e., taking natural logarithms of the regression relationship and then using the linear least-squares estimators for estimates of the parameters. Two explicit types of this model are discussed. In one type it is assumed that the hypothetical regression function is equal to the expected value of Y, and in the other it is assumed that the hypothetical regression function is equal to the median of Y. In the next section, entitled "The Additive Model," it is assumed that the error is normally distributed and is added to the hypothetical regression function. This leads to the problem of least-squares estimation of a nonlinear form.[†] In

[†]In this Memorandum, the two models are also referred to as "logarithmic" (multiplicative error assumed) and "exponential" (additive error assumed).

these two sections, methods of obtaining the estimators, distributions of the estimators, and prediction intervals for the dependent variable, Y, are given. Section IV consists of a practical comparison of the two models, and Section V presents a discussion of the computer program relating to the two models.

Considerable use of matrix notation has been made in the following sections. Hence, familiarity with the ordinary operations of matrix algebra will prove beneficial to the reader. A short discussion of these operations has been included in Appendix A for those who are not familiar with them.

Similarly, some of the statistical facts and relationships that have been assumed in this study are recorded in Appendix B. No attempt has been made to cover all of the statistical ideas assumed in the study, but an attempt has been made to include those necessary ideas that are most easily forgotten. It is suggested that the reader glance through this appendix before proceeding to the main part of the text.

Finally, it should be noticed that e^{μ_0} has been used for the constant multiplier of the regression function instead of an ordinary multiplier, say, $\bar{\mu}_0$. This form has been used purely for mathematical convenience. It should be noted, however, that this imposes the restriction of positivity on the constant multiplier. It has been anticipated that this Memorandum will be used mainly by individuals interested in positive dependent variables such as weight or cost. Hence it is felt that this restriction is of no practical importance. For those who would like a constant (positive) multiplier $\bar{\mu}_0$ in the regression function, instead of e^{μ_0}, a conversion procedure is given in Appendix C.

II. THE MULTIPLICATIVE MODEL

In the multiplicative model it is assumed that for any values of the variables $X_1, X_2, \ldots, X_p$, the corresponding value of Y is given by

$$Y = e^{\mu_0} X_1^{\mu_1} X_2^{\mu_2} \cdots X_p^{\mu_p} \cdot \delta,$$

where δ is a log-normal random variable, i.e.,[†]

$$\ln \delta = \epsilon,$$

and ϵ is a normal random variable with mean and variance given by

$$\text{Med } \epsilon = E\, \epsilon = \nu,$$

$$\text{Var } \epsilon = \sigma_M^2.$$

The mean, variance, and median of δ are then given by[††]

$$E\, \delta = e^{\nu + \sigma_M^2/2}$$

$$\text{Var } \delta = e^{2\nu + \sigma_M^2}\left(e^{\sigma_M^2} - 1\right),$$

$$\text{Med } \delta = e^{\nu}.$$

Notice that the median of δ is not equal to the mean of δ, even though the mean and median of ϵ are identical (since ϵ is a normal random variable and the normal distribution is symmetric).

From these assumptions and because the hypothetical regression

---

[†]The expression "ln" in what follows stands for logarithm to the base e. All logarithms in this Memorandum are taken to the base e.

[††]For further discussion of the log-normal distribution, see [5, p. 89].

function is a constant for any given values of $X_1, X_2, \ldots, X_p$, it follows that Y has a log-normal distribution with

$$E\ Y = \left(e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}\right) e^{\nu + \sigma_M^2/2},$$

$$\text{Var } Y = \left(e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}\right)^2 \cdot e^{2\nu + \sigma_M^2}\left(e^{\sigma_M^2} - 1\right),$$

and

$$\text{Med } Y = \left(e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}\right) e^{\nu}.$$

It also follows that ln Y is a normal random variable with

$$E\ [\ln Y] = \mu_0 + \mu_1 \ln X_1 + \cdots + \mu_p \ln X_p + \nu$$

and

$$\text{Var } [\ln Y] = \sigma_M^2.$$

Two explicit forms of this model will be studied. The first assumes that the hypothetical regression function is the median of the distribution of Y, while the second assumes that the hypothetical regression function is the mean of Y. The choice of which model to use is up to the analyst. It will turn out that this choice affects the estimate of μ_0 and the distribution of this estimate, but fortunately does not affect the other estimates, the distribution of these estimates, or the prediction interval for Y.

ASSUMPTIONS AND IMPLICATIONS OF THE CASES

Median Case

In this case it is assumed that for any given value of the variables $X_1, X_2, \ldots, X_p$, the median of the associated Y is given by

$$\text{Med } Y = e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}.$$

This implies that Med $\delta = 1$, i.e., $e^{\nu} = 1$, and hence that $\nu = 0$. It follows that

$$E\ \delta = e^{\sigma_M^2/2},$$

$$\text{Var}\ \delta = e^{\sigma_M^2}\left(e^{\sigma_M^2} - 1\right),$$

$$E\ Y = e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p} e^{\sigma_M^2/2},$$

and

$$\text{Var}\ Y = \left(e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}\right)^2 \left[e^{\sigma_M^2}\left(e^{\sigma_M^2} - 1\right)\right].$$

It also follows that ln Y is given by

$$\ln Y = \mu_0 + \mu_1 \ln X_1 + \cdots + \mu_p \ln X_p + \epsilon,$$

where ϵ is normally distributed with mean zero and variance σ_M^2. This case has thus reduced to an ordinary linear regression model for ln Y.

Mean Case

In this case it is assumed that for any given values of the variables $X_1, X_2, \ldots, X_p$, the expected value of the associated Y is given by

$$E\ Y = e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}.$$

This implies that $E\ \delta = 1$, i.e., $e^{\nu + \sigma_M^2/2} = 1$, and therefore that $\nu = -\sigma_M^2/2$.

Notice that ν is no longer assumed to be zero. For if $\nu = 0$ were assumed, then, in order to have $E\ \delta = 1$, the assumption that $\sigma_M^2 = 0$ would also have to be made. This would in turn imply that $\text{Var}\ \delta = 0$, or, in other words, that there was no error. But, as mentioned in the Introduction, a model with no error assumption is unrealistic. Hence one cannot assume that $\nu = 0$.

Now from the assumptions of this model it follows that

$$\text{Med } \delta = e^{\nu},$$

$$\text{Med } Y = \left(e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}\right) e^{\nu},$$

$$\text{Var } \delta = e^{\sigma_M^2} - 1,$$

and

$$\text{Var } Y = \left(e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}\right)^2 \left(e^{\sigma_M^2} - 1\right).$$

It also follows that

$$\ln Y = \mu_0 + \mu_1 \ln X_1 + \mu_p \ln X_p + \epsilon,$$

where ϵ is normally distributed with expected value equal to ν and variance equal to σ_M^2. However, since $\nu \neq 0$, this case does not reduce to an ordinary linear regression model for ln Y. The expected value of ln Y is no longer equal to the log of the hypothetical regression function, but is given by

$$E \ln Y = \mu_0 + \mu_1 \ln X_1 + \cdots + \mu_p \ln X_p + \nu.$$

Fortunately, without changing the distribution of ln Y, it is possible to obtain the ordinary linear regression model for ln Y by making a change in one of the parameters. This is accomplished by letting $\mu_0^* = \mu_0 + \nu$ in the model. Then,

$$\ln Y = \mu_0^* + \mu_1 \ln X_1 + \cdots + \mu_p \ln X_p + \epsilon^*,$$

where ϵ^* is normally distributed with mean zero and variance σ_M^2, which is the ordinary linear regression model for ln Y.

Since both cases reduce to essentially the same model, it is clear that problems 1 and 2 posed in the Introduction are essentially the same for the two cases. These cases really differ only in their treatment of the intercept parameter, μ_0.

ESTIMATORS OF THE PARAMETERS

To be able to handle both cases at the same time, let

$$\ln Y = \mu_0^* + \mu_1 \ln X_1 + \cdots + \mu_p \ln X_p + \epsilon^*,$$

where ϵ^* is normally distributed with mean zero and variance σ_M^2. Then μ_0^* and ϵ^* are related to the cases as follows:

	Median Case	Mean Case
$\mu_0^* =$	μ_0	$\mu_0 + \nu$
$\epsilon^* =$	ϵ	$\epsilon - \nu$

Now taking n sets of historical data of the form $(Y_i, X_{1i}, X_{2i}, \ldots, X_{pi})$, it follows that for $i = 1, 2, \ldots, n$,

$$\ln Y_i = \mu_0^* + \mu_1 \ln X_{1i} + \cdots + \mu_p \ln X_{pi} + \epsilon_i^*,$$

where $\epsilon_1^*, \epsilon_2^*, \ldots, \epsilon_n^*$ are independent and identically distributed normal random variables with zero mean and variance equal to σ_M^2. Hence the random vector

$$\begin{bmatrix} \ln Y_1 \\ \ln Y_2 \\ \vdots \\ \ln Y_n \end{bmatrix}$$

has a multivariate normal distribution with mean vector given by

$$\begin{bmatrix} \mu_0^* + \mu_1 \ln X_{11} + \mu_2 \ln X_{21} + \cdots + \mu_p \ln X_{p1} \\ \mu_0^* + \mu_1 \ln X_{12} + \mu_2 \ln X_{22} + \cdots + \mu_p \ln X_{p2} \\ \vdots \\ \mu_0^* + \mu_1 \ln X_{1n} + \mu_2 \ln X_{2n} + \cdots + \mu_p \ln X_{pn} \end{bmatrix}$$

and covariance matrix given by $\sigma_M^2 \cdot I$, where I is the identity matrix of order n.

Now problem 1 in the Introduction was that of picking the "best" estimators of the unknown parameters in the hypothetical regression function. The first problem here, of course, is to define the word "best." By applying the Gauss-Markov theorem to the above multivariate model, it turns out that the maximum likelihood estimators of the parameters are equivalent to the least-squares estimators of the parameters and that among the class of unbiased, linear estimates they have the property that their variances are simultaneously smallest. Hence the least-squares estimators seem to be reasonable candidates for the definition of "best" estimators, and this definition will be adopted in the present section.

Before solving for the least-squares estimators, it is desirable to change the form of the equations slightly. The reason for this will become apparent when the distribution of the estimators is discussed in the next section. So for $i = 1, 2, \ldots, n$, let

$$\ln Y_i = \mu_0' + \mu_1(\ln X_{1i} - \overline{\ln X}_1) + \cdots + \mu_p(\ln X_{pi} - \overline{\ln X}_p) + \epsilon^*,$$

where

$$\overline{\ln X}_j = \sum_{k=1}^{n} \frac{\ln X_{jk}}{n}$$

for $j = 1, 2, \ldots, p$. The least-squares estimators of $\mu_1, \ldots, \mu_p$ for this form will be the same as those for the previous form. Of course the least-squares estimator of μ_0' will be obtained instead of the least-squares estimator of μ_0^*. However, the estimator for the latter can be found by using the relationship

$$\mu_0^* = \mu_0' - \mu_1 \overline{\ln X}_1 - \mu_2 \overline{\ln X}_2 - \cdots - \mu_p \overline{\ln X}_p.$$

Hence no restrictions have been imposed by this change.

The following notation will also prove helpful. Let S be a p x p symmetric matrix whose elements are given by

$$s_{ij} = \begin{cases} \sum_{k=1}^{n} (\ln X_{ik} - \overline{\ln X}_i)^2 & \text{if} \quad i = j, \\ \sum_{k=1}^{n} (\ln X_{ik} - \overline{\ln X}_i)(\ln X_{jk} - \overline{\ln X}_j) & \text{if} \quad i \neq j, \end{cases}$$

for $i = 1, 2, \ldots, p$; $j = 1, 2, \ldots, p$. Let T be the p-dimensional column vector whose elements are given by

$$t_i = \sum_{k=1}^{n} (\ln X_{ik} - \overline{\ln X}_i)(\ln Y_k - \overline{\ln Y})$$

for $i = 1, 2, \ldots, p$.

The least-squares estimators of $\mu_0', \mu_1, \ldots, \mu_p$ are those values of these parameters that minimize

$$Q(\mu_0', \mu_1, \ldots, \mu_p) = \sum_{k=1}^{n} \{\ln Y_k - [\mu_0' + \mu_1(\ln X_{1k} - \overline{\ln X}_1) + \cdots + \mu_p(\ln X_{pk} - \overline{\ln X}_p)]\}^2.$$

By taking the partial derivatives, $(\partial Q)/(\partial \mu_0')$ and $(\partial Q)/(\partial \mu_i)$, $i = 1, 2, \ldots, p$, and setting them equal to zero, the following $p + 1$ equations are obtained:

$$\mu_0' = \overline{\ln Y} - \left(\frac{1}{n}\right) \cdot \left[\sum_{k=1}^{n} \mu_1(\ln X_{1k} - \overline{\ln X_1}) + \cdots + \mu_p (\ln X_{pk} - \overline{\ln X_p})\right]$$

$$= \overline{\ln Y}$$

and $s_{i1}\mu_1 + s_{i2}\mu_2 + \cdots + s_{ip}\mu_p = t_i$ for $i = 1, 2, \ldots, p$.

In matrix notation these equations are given by

$$\mu_0' = \overline{\ln Y}$$

and

$$S \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{bmatrix} = T.$$

The least-squares estimators of the parameters are then given by the values of the parameters that satisfy the above equations. Hence these estimators are given by

$$\hat{\mu}_0' = \overline{\ln Y}$$

and

$$\begin{bmatrix} \hat{\mu}_1 \\ \hat{\mu}_2 \\ \vdots \\ \hat{\mu}_p \end{bmatrix} = S^{-1}T.$$

The least-squares estimator of μ_0^* is then given by

$$\hat{\mu}_0^* = \hat{\mu}_0' - \hat{\mu}_1 \overline{\ln X_1} - \cdots - \hat{\mu}_p \overline{\ln X_p}.$$

All the estimators for the median case have thus been obtained, since $\mu_0 = \mu_0^*$ for this case and, therefore, $\hat{\mu}_0 = \hat{\mu}_0^*$. But in the mean case, $\mu_0^* = \mu_0 + \nu$ and $\nu = -\sigma_M^2/2$. To obtain an estimator of μ_0, an estimator of ν, and hence σ_M^2, must first be obtained.

The estimator of σ_M^2 that will be used is given by

$$\hat{\sigma}_M^2 = \sum_{k=1}^{n} \frac{(\ln Y_k - \widehat{\ln Y_k})^2}{n - (p + 1)},$$

where $\widehat{\ln Y_k} = \hat{\mu}_0' + \hat{\mu}_1(\ln X_{1k} - \overline{\ln X_1}) + \cdots + \hat{\mu}_p(\ln X_{pk} - \overline{\ln X_p})$

$$= \hat{\mu}_0^* + \hat{\mu}_1 \ln X_{1k} + \cdots + \hat{\mu}_p \ln X_{pk} \quad \text{for} \quad k = 1, 2, \ldots, n.$$

There are three reasons for choosing this estimator:

(i) $\hat{\sigma}_M^2$ is the maximum likelihood estimate of σ_M^2.

(ii) $\hat{\sigma}_M^2$ is an unbiased estimate of σ_M^2, i.e., $E\,\hat{\sigma}_M^2 = \sigma_M^2$.

(iii) $\hat{\sigma}_M^2$ is independent of all the least-squares estimators of the μ's, previously obtained in this section.

Now let the estimators of ν and μ_0 be given by $\hat{\nu} = -\hat{\sigma}_M^2/2$ and $\hat{\mu}_0 = \hat{\mu}_0^* - \hat{\nu}$, respectively. Notice that with these estimators it follows that for $k = (1, 2, \ldots, n)$, $\widehat{\ln Y_k}$, the estimate of $E \ln Y_k$, is given by

$$\widehat{\ln Y_k} = \begin{cases} \hat{\mu}_0 + \hat{\mu}_1 \ln X_{1k} + \cdots + \hat{\mu}_p \ln X_{pk} & \text{in the median case,} \\ \hat{\mu}_0 + \hat{\mu}_1 \ln X_{1k} + \cdots + \hat{\mu}_p \ln X_{pk} + \hat{\nu} & \text{in the mean case.} \end{cases}$$

Two other things should be noted before concluding this section. First, it is obvious that the median case of the multiplicative model

is the usual model that one uses to fit a multiple linear regression function to a set of historical data. That is, when one takes logarithms of the data and fits the parameters via least squares, one is implicitly assuming the median case of the multiplicative model.

Second, the estimation of $\hat{\sigma}_M^2$ indicates that the inequality $n > p + 1$ must be satisfied, or else $\hat{\sigma}_M^2$ would be negative or undefined, which is clearly not desirable. Another reason for this restriction can be found by examining the S matrix. If the condition that $n \geq p + 1$ is not satisfied, then S will be singular and hence S^{-1} will not exist. Since this restriction will also be necessary in the additive model, the assumption that $n > p + 1$ will be made throughout.

DISTRIBUTION OF THE ESTIMATORS

Before solving problem 2 in the Introduction, it will be necessary to find out some facts about the distributions of the estimators. We will need the expected values, variances, covariances, and (at least approximately) the distributions themselves of each of the estimators.

Looking first at the estimators $\hat{\mu}_1, \ldots, \hat{\mu}_p$, it is clear from the equations

$$\begin{bmatrix} \hat{\mu}_1 \\ \vdots \\ \hat{\mu}_p \end{bmatrix} = S^{-1}T$$

that each of these estimators is a linear combination of the normal random variables $\ln Y_i$. Hence the estimators are also normally distributed. It has already been stated that the estimators are unbiased, i.e.,

$$E\,\hat{\mu}_i = \mu_i \quad \text{for} \quad i = 1, 2, \ldots, p.$$

Finally it can be shown that the covariance matrix of these estimators is given by $\sigma_M^2 S^{-1}$.† Therefore we have that

$$\text{Cov}\,(\hat{\mu}_i, \hat{\mu}_j) = \sigma_M^2 s^{ij}, \quad i \neq j,\ i = 1, 2, \ldots, p;\ j = 1, 2, \ldots, p,$$

and

$$\text{Var}\,\hat{\mu}_i = \sigma_M^2 s^{ii}, \quad i = 1, 2, \ldots, p,$$

where s^{ij} is the (i, j)-th element of S^{-1}. Of course, σ_M^2 is unknown, but for estimates of these variances and covariances one may replace σ_M^2 by $\hat{\sigma}_M^2$.

It should be noted that the simple form of the covariance matrix was obtained as a consequence of the change in the intercept of the model from μ_0^* to μ_0', as discussed earlier. Indeed, the reason for that change was to facilitate this nice form of the covariance matrix.

Next, consider the distribution of $\hat{\mu}_0'$. It also is normally distributed and unbiased for the same reasons that were given in the previous argument. Furthermore, being the sample mean of the $\ln Y_i$'s, its variance is given by σ_M^2/n. It also turns out that $\hat{\mu}_0'$ is independent of the other least-squares estimators, a fact that can be proved by an algebraic argument that uses orthogonal projections. Hence,

$$E\,\hat{\mu}_0' = \mu_0',$$

$$\text{Var}\,\hat{\mu}_0' = \frac{\sigma_M^2}{n},$$

$$\text{Cov}\,(\hat{\mu}_0', \hat{\mu}_i) = 0 \quad \text{for} \quad i = 1, 2, \ldots, p.$$

Again σ_M^2 can be estimated by $\hat{\sigma}_M^2$.

---

†For proof, see [1, pp. 245-250].

As mentioned earlier, $\hat{\sigma}_M^2$ is independent of all of the estimators of the μ's mentioned so far in this section. Thus the covariances of $\hat{\sigma}_M^2$ with the $\hat{\mu}_i$'s are zero. Looking at the equation for $\hat{\sigma}_M^2$, one sees that $[n - (p + 1)]\hat{\sigma}_M^2$ is equal to the sum of squares of independent normal random variables that have zero mean and variance σ_M^2. Hence,

$$\frac{\hat{\sigma}_M^2[n - (p + 1)]}{\sigma_M^2}$$

has a chi-square distribution with $n - (p + 1)$ degrees of freedom. The loss of $p + 1$ degrees of freedom is due to the $p + 1$ linear relationships already established between the random variables, $\ln Y_i$, by the least-squares equations

$$\mu_0' = \overline{\ln Y}$$

and

$$S \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_p \end{bmatrix} = T.$$

Now since the expected value of a chi-square random variable is equal to its degrees of freedom, and its variance is equal to twice its degrees of freedom, it is easy to see that

$$E\ \hat{\sigma}_M^2 = \sigma_M^2$$

and

$$\text{Var}\ \hat{\sigma}_M^2 = \frac{2(\sigma_M^2)^2}{[n - (p + 1)]}.$$

The distribution of $\hat{\mu}_0^*$ is, of course, normal, as $\hat{\mu}_0^*$ is a linear combination of normally distributed variables, namely,

$$\hat{\mu}_0^* = \hat{\mu}_0' - \hat{\mu}_1 \overline{\ln X}_1 - \cdots - \hat{\mu}_p \overline{\ln X}_p.$$

The expected value is given by

$$\begin{aligned} E\,\hat{\mu}_0^* &= E\,\hat{\mu}_0' - \overline{\ln X}_1 E\,\hat{\mu}_1 - \cdots - \overline{\ln X}_p E\,\hat{\mu}_p \\ &= \mu_0' - \overline{\ln X}_1 \mu_1 - \cdots - \overline{\ln X}_p \mu_p = \mu_0^*. \end{aligned}$$

Hence $\hat{\mu}_0^*$ is also unbiased. The variance and covariances are found by the following:

$$\begin{aligned} \text{Var}\,\hat{\mu}_0^* &= \text{Var}\,\hat{\mu}_0' - 2\sum_{i=1}^{p} \text{Cov}\,(\hat{\mu}_0', \hat{\mu}_i)\,\overline{\ln X}_i \\ &\quad + \sum_{i=1}^{p} \left(\overline{\ln X}_i^{\,2}\,\text{Var}\,\hat{\mu}_i + \sum_{\substack{j=1 \\ j\neq i}}^{p} \overline{\ln X}_i\,\overline{\ln X}_j\,\text{Cov}\,(\hat{\mu}_i, \hat{\mu}_j) \right) \\ &= \sigma_M^2/n + \sigma_M^2 \sum_{i=1}^{p} \overline{\ln X}_i \left(\sum_{j=1}^{p} \overline{\ln X}_j\, s^{ij} \right) \\ &= \sigma_M^2/n + \sigma_M^2 (\overline{\ln X}_1, \ldots, \overline{\ln X}_p) S^{-1} \begin{bmatrix} \overline{\ln X}_1 \\ \vdots \\ \overline{\ln X}_p \end{bmatrix}. \end{aligned}$$

For $i = 1, 2, \ldots, p$,

$$\begin{aligned} \text{Cov}\,(\hat{\mu}_0^*, \hat{\mu}_i) &= E\,\hat{\mu}_0^*\hat{\mu}_i - E\,\hat{\mu}_0^*\,E\,\hat{\mu}_i \\ &= E\,(\hat{\mu}_0'\hat{\mu}_i - \hat{\mu}_i\hat{\mu}_1 \overline{\ln X}_1 - \cdots - \hat{\mu}_i\hat{\mu}_p \overline{\ln X}_p) \\ &\quad - E\,\hat{\mu}_i\,E\,(\hat{\mu}_0' - \hat{\mu}_1 \overline{\ln X}_1 - \cdots - \hat{\mu}_p \overline{\ln X}_p) \\ &= E\,\hat{\mu}_0'\hat{\mu}_i - E\,\hat{\mu}_i\,E\,\hat{\mu}_0' \\ &\quad - \left(\sum_{j=1}^{p} (E\,\hat{\mu}_i\hat{\mu}_j - E\,\hat{\mu}_i\,E\,\hat{\mu}_j)\,\overline{\ln X}_j \right) \end{aligned}$$

$$= \text{Cov}\,(\hat{\mu}_0', \hat{\mu}_i) - \overline{\ln X}_i \,\text{Var}\,\hat{\mu}_i - \sum_{\substack{j=1 \\ j\neq i}}^{p} \overline{\ln X}_j \,\text{Cov}\,(\hat{\mu}_i, \hat{\mu}_j)$$

$$= -\sigma_M^2 \sum_{j=1}^{p} \overline{\ln X}_j \, s^{ij}.$$

Therefore,

$$\begin{bmatrix} \text{Cov}\,(\hat{\mu}_0^*, \hat{\mu}_1) \\ \text{Cov}\,(\hat{\mu}_0^*, \hat{\mu}_2) \\ \vdots \\ \text{Cov}\,(\hat{\mu}_0^*, \hat{\mu}_p) \end{bmatrix} = -\sigma_M^2 S^{-1} \begin{bmatrix} \overline{\ln X}_1 \\ \overline{\ln X}_2 \\ \vdots \\ \overline{\ln X}_p \end{bmatrix}.$$

The distributions of all the estimators for the median case have now been obtained since $\hat{\mu}_0 = \hat{\mu}_0^*$ in this case. In the mean case, however, the distribution of $\hat{\mu}_0$ is still to be determined. For this case, $\hat{\mu}_0 = \hat{\mu}_0^* - \hat{v} = \hat{\mu}_0^* - \hat{\sigma}_M^2/2$.

The exact distribution of $\hat{\mu}_0$ is not really obtainable since it is a linear combination of a normal and (essentially) a chi-square random variable. For large n, however, it can be assumed that $\hat{\mu}_0$ is approximately normal. An easier problem is that of determining the mean and variance of $\hat{\mu}_0$ and the covariances of $\hat{\mu}_0$ and $\hat{\mu}_i$ for i = 1, 2, ..., p. Since $\hat{\sigma}_M^2$ is independent of $\hat{\mu}_0'$, $\hat{\mu}_1$, ..., $\hat{\mu}_p$ and since $\hat{\mu}_0^*$ is a function of only these random variables, it follows that $\hat{\sigma}_M^2$ is independent of $\hat{\mu}_0^*$. Therefore,

$$E\,\hat{\mu}_0 = E\,\hat{\mu}_0^* - \tfrac{1}{2}E\,\hat{\sigma}_M^2 = \mu_0^* - \tfrac{1}{2}\sigma_M^2 = \mu_0,$$

$$\text{Var } \hat{\mu}_0 = \text{Var } \hat{\mu}_0^* + \frac{1}{4} \text{Var } \hat{\sigma}_M^2,$$

and

$$\text{Cov } (\hat{\mu}_0, \hat{\mu}_i) = \text{Cov } (\hat{\mu}_0^*, \hat{\mu}_i) \quad \text{for} \quad i = 1, 2, \ldots, p.$$

The distributions of all of the estimators of the mean case have thus been obtained. We are now able to attack problem 2 of the Introduction.

PREDICTION INTERVALS

One of the problems posed in the Introduction is that of providing a measure of how well Y can be predicted by the estimated regression function. A prediction interval will be used for this purpose. This takes the form of

$$P\{Y_\ell < Y < Y_u\} = \gamma,$$

where $0 < \gamma < 1$, and Y_ℓ and Y_u are bounds that depend on γ and the values of the variables $X_1, X_2, \ldots, X_p$.[†] The prediction interval takes into account the variability of the predicted Y due to the random nature of the estimates of the parameters of the regression function as well as that due to the error assumption in the hypothetical regression function. For a given γ, the predictive ability is, of course, best for those values of $X_1, \ldots, X_p$ for which the interval is smallest. Notice that the prediction interval also gives high and low estimates of Y for each γ.

To obtain the bounds, again examine the predicted value of E ln Y as expressed by the following:

$$(1) \qquad \widehat{\ln Y} = \hat{\mu}_0^* + \hat{\mu}_1 \ln X_1 + \cdots + \hat{\mu}_p \ln X_p$$

[†]These values will not generally be from the sample.

$$(2) \qquad = \hat{\mu}_0' + \hat{\mu}_1 (\ln X_1 - \overline{\ln X}_1) + \cdots + \hat{\mu}_p (\ln X_p - \overline{\ln X}_p)$$

$$(3) \qquad = \begin{cases} \hat{\mu}_0 + \hat{\mu}_1 \ln X_1 + \cdots + \hat{\mu}_p \ln X_p & \text{(median case)} \\ \hat{\mu}_0 + \hat{\mu}_1 \ln X_1 + \cdots + \hat{\mu}_p \ln X_p + \hat{v} & \text{(mean case)}. \end{cases}$$

In these equations, $\overline{\ln X}_1, \ldots, \overline{\ln X}_p$ are the same quantities used in the estimation procedure. They are functions of the historical data only. On the other hand $X_1, \ldots, X_p$ or $\ln X_1, \ldots, \ln X_p$ are the values of the independent variables for the Y that is being predicted.

From Eq. (1) or (2) it is easy to see that $\widehat{\ln Y}$ is normally distributed, being a linear combination of normals, with

$$E \widehat{\ln Y} = \mu_0^* + \mu_1 \ln X_1 + \cdots + \mu_p \ln X_p$$

and

$$\text{Var } \widehat{\ln Y} = \frac{\sigma_M^2}{n} + \sigma_M^2 [(\ln X_1 - \overline{\ln X}_1), \ldots, (\ln X_p - \overline{\ln X}_p)] S^{-1} \begin{bmatrix} \ln X_1 - \overline{\ln X}_1 \\ \vdots \\ \ln X_p - \overline{\ln X}_p \end{bmatrix}.$$

(To derive Var $\widehat{\ln Y}$ use (2) and the same reasoning as that used earlier to obtain Var $\hat{\mu}_0^*$.)

From the assumptions of this model, it was seen that ln Y was normally distributed with

$$E \ln Y = \mu_0^* + \mu_1 \ln X_1 + \cdots + \mu_p \ln X_p$$

and Var ln Y = σ_M^2. Furthermore, it is independent of $\widehat{\ln Y}$ since all the randomness of ln Y comes from ϵ^*. Hence the random variable D, defined by

$$D = \ln Y - \widehat{\ln Y}$$

is normally distributed with

$$E\ D = 0 \quad \text{and} \quad \text{Var}\ D = \sigma_M^2 V,$$

where

$$V = 1 + 1/n + [(\ln X_1 - \overline{\ln X_1}), \ldots, (\ln X_p - \overline{\ln X_p})]S^{-1} \begin{bmatrix} \ln X_1 - \overline{\ln X_1} \\ \vdots \\ \ln X_p - \overline{\ln X_p} \end{bmatrix}.$$

A prediction interval can now be set up using the standard normal random variable

$$\frac{D}{\sqrt{\sigma_M^2 V}}.$$

To get numerical bounds, however, σ_M^2 would have to be estimated by $\hat{\sigma}_M^2$. A more accurate prediction interval, which takes into account this estimation, can be set up as follows. It has previously been shown that

$$\frac{[n - (p + 1)]\hat{\sigma}_M^2}{\sigma_M^2}$$

has a chi-square distribution with $n - (p + 1)$ degrees of freedom. Furthermore it is independent of $\ln Y$ and $\widehat{\ln Y}$ (seen from Eq. (2)) and D, and therefore it is independent of

$$\frac{D}{\sqrt{\sigma_M^2 V}}.$$

Hence,

$$\frac{D/\sqrt{V\sigma_M^2}}{\sqrt{\frac{[n - (p + 1)]\hat{\sigma}_M^2/\sigma_M^2}{n - (p + 1)}}} = \frac{D}{\sqrt{V\hat{\sigma}_M^2}}$$

has a t-distribution with $n - (p + 1)$ degrees of freedom.

To obtain the prediction interval then, choose the probability, γ, that is to be covered by the interval. Let $\alpha = (1 + \gamma)/2$. Then find the $\alpha(100)$-th percentile of the t-distribution with $n - (p + 1)$ degrees of freedom. Call this t_α. Since the t-distribution is symmetric about zero, we have $t_{1-\alpha} = -t_\alpha$. Therefore,

$$P\left\{t_{1-\alpha} < \frac{D}{\sqrt{V\hat{\sigma}_M^2}} < t_\alpha\right\} = \gamma$$

or

$$P\left\{-t_\alpha \sqrt{V\hat{\sigma}_M^2} < D < t_\alpha \sqrt{V\hat{\sigma}_M^2}\right\} = \gamma$$

or

$$P\left\{\widehat{\ln Y} - t_\alpha \sqrt{V\sigma_M^2} < \ln Y < \widehat{\ln Y} + t_\alpha \sqrt{V\hat{\sigma}_M^2}\right\} = \gamma.$$

The latter is a prediction interval on ln Y. But since exponentiation is a monotone function, it can be performed inside the brackets without altering the probability content of the interval. Therefore the prediction interval for Y is given by

$$P\left\{e^{\widehat{\ln Y} - t_\alpha \sqrt{V\hat{\sigma}_M^2}} < Y < e^{\widehat{\ln Y} + t_\alpha \sqrt{V\hat{\sigma}_M^2}}\right\} = \gamma.$$

Notice that this interval is the same for both cases. To describe the interval in terms of the estimated regression function of the cases, it follows from Eq. (3) that

$$e^{\widehat{\ln Y}} = \begin{cases} e^{\hat{\mu}_0 X_1^{\hat{\mu}_1} \cdots X_p^{\hat{\mu}_p}} & \text{for the median case,} \\ \left(e^{\hat{\mu}_0 X_1^{\hat{\mu}_1} \cdots X_p^{\hat{\mu}_p}}\right) e^{\hat{\nu}} & \text{for the mean case.} \end{cases}$$

Hence the prediction intervals are given by

$$P\left\{\hat{Y}e^{-t_{\alpha}\sqrt{V\hat{\sigma}_M^2}} < Y < \hat{Y}e^{t_{\alpha}\sqrt{V\sigma_M^2}}\right\} = \gamma$$

for the median case and by

$$P\left\{\hat{Y}e^{\hat{\nu}}e^{-t_{\alpha}\sqrt{V\hat{\sigma}_M^2}} < Y < \hat{Y}e^{\hat{\nu}}e^{t_{\alpha}\sqrt{V\hat{\sigma}_M^2}}\right\} = \gamma$$

for the mean case, where $\hat{Y}$ is the predicted value of Y given by

$$\hat{Y} = e^{\hat{\mu}_0} X_1^{\hat{\mu}_1} \cdots X_p^{\hat{\mu}_p}.$$

Again it should be noted that the prediction interval does not depend on the particular case being used. The endpoints of the interval look different because $\hat{\mu}_0$, and therefore $\hat{Y}$, is different for the two cases (i.e., $\hat{\mu}_0$ (mean case) = $\hat{\mu}_0$ (median case) - $\hat{\nu}$). It is the positioning of the estimated regression function in the prediction interval that is different, and not the prediction interval itself.

This concludes the discussion of the multiplicative model.

III. THE ADDITIVE MODEL

GENERAL MODEL

In the additive model it is assumed that for any given values of $X_1, X_2, \ldots, X_p$, the value of Y is given by

$$Y = e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p} + \eta,$$

where η is a normally distributed random variable. Since it is usually considered desirable to have the distribution of Y centered on the regression function, it is assumed that the mean of η is zero. The variance of η will be denoted by σ_A^2 and is assumed to be unknown. Hence Y itself is normally distributed with

$$E\ Y = \text{Med}\ Y = e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}$$

and

$$\text{Var}\ Y = \sigma_A^2.$$

Notice that the variance of Y is independent of the value of $X_1, \ldots, X_p$. Notice also that the median of Y is the same as the mean of Y; hence in this model we only have to consider the one case.

The distribution of the historical data, which consists of n sets of the form $(Y_i, X_{1i}, X_{2i}, \ldots, X_{pi})$, where $n > p + 1$, is easily obtainable from the above.

The random vector

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}$$

has a multivariate normal distribution with mean vector given by

$$\begin{bmatrix} e^{\mu_0} X_{11}^{\mu_1} X_{21}^{\mu_2} \cdots X_{p1}^{\mu_p} \\ e^{\mu_0} X_{12}^{\mu_1} X_{22}^{\mu_2} \cdots X_{p2}^{\mu_p} \\ \vdots \\ e^{\mu_0} X_{1n}^{\mu_1} X_{2n}^{\mu_2} \cdots X_{pn}^{\mu_p} \end{bmatrix}$$

and covariance matrix given by $\sigma_A^2 \cdot I$, where I is the n-dimensional identity matrix.

ESTIMATION

To tackle problem 1 of the Introduction, it is again necessary to determine what is meant by "best" estimators. Of course "best" may be defined in many ways. One way that has some nice properties is that of using the least-squares estimators as the "best" estimators. That is, choose as estimators of $\mu_0, \mu_1, \ldots, \mu_p$ those values that minimize

$$Q(\mu_0, \mu_1, \ldots, \mu_p) = \sum_{i=1}^{n} \left(Y_i - e^{\mu_0} X_{1i}^{\mu_1} \cdots X_{pi}^{\mu_p}\right)^2.$$

Besides being the least-squares estimators, these estimators are the maximum-likelihood estimators and are also approximately unbiased. Because of these properties, "best" estimators will be defined as least-squares estimators in this section.

The problem of finding these estimators will be much more difficult in this case than it was in the multiplicative case (where a linear regression on ln Y was assumed) since nonlinear least-squares estimators

must be obtained. To see why problems arise let us first attack the problem directly. The usual way to minimize Q would be to take the partial derivatives of Q with respect to $\mu_0, \mu_1, \ldots, \mu_p$ and set them equal to zero. After doing this, the following least-squares equations are obtained:

$$\frac{\partial Q}{\partial \mu_0} = -2 \sum_{i=1}^{n} (Y_i - e^{\mu_0 X_{1i}^{\mu_1} \cdots X_{pi}^{\mu_p}}) e^{\mu_0 X_{1i}^{\mu_1} \cdots X_{pi}^{\mu_p}} = 0,$$

$$\frac{\partial Q}{\partial \mu_j} = -2 \sum_{i=1}^{n} (Y_i - e^{\mu_0 X_{1i}^{\mu_1} \cdots X_{pi}^{\mu_p}}) e^{\mu_0 X_{1i}^{\mu_1} \cdots X_{pi}^{\mu_p}} \ln X_{ji} = 0$$

$$\text{for } j = 1, 2, \ldots, p.$$

Having obtained these equations, one would try to solve them for $\mu_0, \ldots, \mu_p$. But this leads to the following three problems:

(i) How does one solve the $p + 1$ nonlinear equations for the $p + 1$ unknowns?

(ii) Having accomplished (i), is this solution unique?

(iii) Having accomplished (i), is this solution a minimum?

Problems (ii) and (iii) arise from a generalization of the problem of solving the equation

$$\frac{df(X)}{dX} = 0,$$

where f(X) is a function of just one variable X. It is not true for every function f that there exists just one value of X for which the derivative, $[df(X)]/(dX)$, is equal to zero. Any such value is called a critical point and can represent a relative or absolute maximum or minimum, or an inflection point of the function f(X). In like manner the set of least-squares equations can have more than one solution.

Each solution is a critical point of the equations and may represent a relative minimum or maximum, an absolute minimum, or a saddle point in the surface described by $Q(\mu_0, \mu_1, \ldots, \mu_p)$. (From the nature of Q itself it is fairly clear that no finite critical point will represent an absolute maximum.) Hence care must be taken in the minimization of Q.

It should be noted that these problems do not arise in the multiplicative model, since in that case we have p + 1 linear equations in p + 1 unknowns and therefore there is a unique solution for the parameters that minimizes Q (assuming that the equations are linearly independent, i.e., that the determinant of S is not equal to zero).

Hartley [3] has studied the problems posed in (i), (ii), and (iii) for general nonlinear regression functions. The solutions to the questions given in this section are a specialization of his results. However, a slight modification of his convergence routine is used here, so the following is not a direct specialization.

To find a solution for the least-squares equations, first make an initial choice for the values of the parameters. Call these $\mu_0^0, \mu_1^0, \ldots, \mu_p^0$.

Use these values to evaluate the following p + 1 equations:

$$\sum_{i=1}^{n} \left\{ \left(e^{\mu_0^0 X_{1i}^{\mu_1^0}} \cdots X_{pi}^{\mu_p^0} \right)^2 \left(D_0 + \sum_{j=1}^{p} \ln X_{ji} D_j \right) \right\}$$

$$= \left(\sum_{i=1}^{n} \left(Y_i - e^{\mu_0^0 X_{1i}^{\mu_1^0}} \cdots X_{pi}^{\mu_p^0} \right) e^{\mu_0^0 X_{1i}^{\mu_1^0}} \cdots X_{pi}^{\mu_p^0} \right),$$

$$\sum_{i=1}^{n} \left\{ \left(e^{\mu_0^0 \mu_1^0} X_{1i}^{\mu_1^0} \cdots X_{pi}^{\mu_p^0} \right)^2 \left(D_0 \ln X_{ki} + \sum_{j=1}^{p} \ln X_{ki} \ln X_{ji} D_j \right) \right.$$

$$= \left(\sum_{i=1}^{n} \left(Y_i - e^{\mu_0^0 \mu_1^0} X_{1i}^{\mu_1^0} \cdots X_{pi}^{\mu_p^0} \right) e^{\mu_0^0 \mu_1^0} X_{1i}^{\mu_1^0} \cdots X_{pi}^{\mu_p^0} \ln X_{ki} \right)$$

for $k = 1, 2, \ldots, p$.

Notice that we now have $p + 1$ linear equations in $p + 1$ unknowns, $D_0, \ldots, D_p$. These equations correspond to the equations given in (10) of Hartley's paper. Solve the equations for $D_0, D_1, \ldots, D_p$. Next, new values for the parameters, say, $\mu_0^1, \mu_1^1, \ldots, \mu_p^1$, must be selected.

Hartley gives a method of doing this for which the following relation holds:

$$Q\left(\mu_0^0, \mu_1^0, \ldots, \mu_p^0\right) \geq Q\left(\mu_0^1, \mu_1^1, \ldots, \mu_p^1\right).$$

For computing purposes, Hartley's method has been altered, but the above relationship still holds. In the computer method, the function Q is calculated as a function of a single parameter ℓ, namely,

$$Q\left(\mu_0^0 + \ell D_0, \mu_1^0 + \ell D_1, \ldots, \mu_p^0 + \ell D_p\right),$$

for $\ell = 0, 0.1, 0.2, \ldots, 1.0$. Let ℓ^0 equal the value of ℓ for which Q_ℓ is minimized among $\ell = 0, 0.1, \ldots, 1.0$. If $\ell^0 = 0$, calculate Q_ℓ for $\ell = 0, 0.01, \ldots, 0.1$. Recalculate ℓ^0 reducing the values of ℓ by a factor of 0.1 each iteration, until $\ell^0 \neq 0$ (or until the values of ℓ get smaller than some specified number †). If ℓ^0 is not equal to

†In this case consider the iterative process finished.

zero, calculate μ_k^1 by

$$\mu_k^1 = \mu_k^0 + \ell^0 D_k \quad \text{for} \quad k = 0, 1, \ldots, p.$$

Now use the set $\left(\mu_0^1, \mu_1^1, \ldots, \mu_p^1\right)$ in place of $\left(\mu_0^0, \mu_1^0, \ldots, \mu_p^0\right)$ in the linear equations to calculate a new $D_0, D_1, \ldots, D_p$. From this, calculate a new set $\left(\mu_0^2, \ldots, \mu_p^2\right)$ as above and continue this process until the μ_k^t's differ from the μ_k^{t-1}'s for all k by less than some specific value. Let the resulting values of the set $\left(\mu_0^t, \ldots, \mu_p^t\right)$ be denoted by $\left(\hat{\mu}_0, \hat{\mu}_1, \ldots, \hat{\mu}_p\right)$ and end the iterative process.

The resulting values of the parameters are (approximate) solutions to the least-squares equations. Moreover, convergence is always obtained, and the solution represents at least a relative minimum for Q. This is because of the procedure used for picking the new values of $\mu_0, \mu_1, \ldots, \mu_p$ at each stage of the iteration. At each stage the inequality

$$Q\left(\mu_0^{t-1}, \ldots, \mu_p^{t-1}\right) \geq Q\left(\mu_0^t, \mu_1^t, \ldots, \mu_p^t\right)$$

holds; and therefore as $\left(\mu_0^t, \ldots, \mu_p^t\right)$ converges to $\left(\hat{\mu}_0, \ldots, \hat{\mu}_p\right)$, Q converges to a relative minimum.

The assertions that the solution for the least-squares equations obtained is unique and is an absolute minimum for Q are harder to establish. In his paper, Hartley discusses a sufficient condition for these assertions. It is that the matrix of second partial derivatives of Q, with respect to $\mu_0, \ldots, \mu_p$, be postive definite for all values of the parameters. In general this will not be true. In fact for the regression function discussed in this study, it has not been proved, but it can be aserted that Q is minimized over the largest connected

region of the parameter space containing $(\hat{\mu}_0, \ldots, \hat{\mu}_p)$ and having the property that the second partial derivative matrix is postive definite when evaluated at any $(\mu_0, \ldots, \mu_p)$ in the region. It will be positive definite when evaluated at $(\hat{\mu}_0, \ldots, \hat{\mu}_p)$, since this point represents at least a relative minimum. Furthermore, $(\hat{\mu}_0, \ldots, \hat{\mu}_p)$ will be the only solution to the least-squares equations in this region.

For the case under consideration, it can be shown that if

$$2e^{\mu_0} X_{1i}^{\mu_1} \cdots X_{pi}^{\mu_p} - Y_i > 0 \quad \text{for} \quad i = 1, 2, \ldots, n,$$

then the matrix of second partial derivatives of Q is postive definite. Hence if this holds for $(\hat{\mu}_0, \ldots, \hat{\mu}_p)$, then the estimators represent an absolute minimum of Q and the only critical point for Q in the region of the parameters $(\mu_0, \ldots, \mu_p)$ that satisfy the above inequalities. If the inequalities do not hold for $(\hat{\mu}_0, \ldots, \hat{\mu}_p)$, one should take initial estimates of the parameters from the above region and obtain the solution in the region. For solutions outside the region, no general argument can be given at this time for proving uniqueness or absolute minimum properties on some other region. (For further discussion of the sufficiency of a postive definite matrix of second partials and for a proof of positive definiteness of the above region, see Appendix D.)

It should be noted that it seems to be a reasonable assumption that the hypothetical regression function lies in the region described above. For it seems that if the relationship between Y and $X_1 \ldots, X_p$ being studied here is valid, then the historical data should satisfy $2e^{\mu_0} X_{1i}^{\mu_1} \cdots X_{pi}^{\mu_p} - Y_i > 0$ for $i = 1, 2, \ldots, p$. This should certainly

be true when Y is not too close to zero. Hence it seems reasonable to establish the criterion

$$2e^{\hat{\mu}_0} X_{1i}^{\hat{\mu}_1} \cdots X_{pi}^{\hat{\mu}_p} - Y_i > 0 \quad \text{for } i = 1, 2, \ldots, n$$

as a necessary condition for a good fit.

DISTRIBUTION OF THE ESTIMATORS

As in the multiplicative model, before answering problem 2 of the Introduction, the distribution of the estimators must be found. Just as the solution for the estimators was more difficult for this case than for the multiplicative case, so also will the finding of the distribution of the estimators be more difficult, and only approximations can be obtained. Gabler and Madansky [2] have studied this problem for the general least-squares regression model. The equations utilized in this study represent a direct specialization of their model.

Because of the nature of the least-squares estimating procedure, the estimators $(\hat{\mu}_0, \hat{\mu}_1, \ldots, \hat{\mu}_p)$ are approximately normally distributed and approximately unbiased, i.e.,

$$E\,\hat{\mu}_k \approx \mu_k \quad \text{for} \quad k = 0, 1, 2, \ldots, p.$$

The word "approximate" must be used in two senses. First the estimates represent only an approximate solution to the least-squares equations due to the iterative solution for these estimators. Also, they are only approximately unbiased because of the nonlinear form of the regression equation.

By using the Gabler-Mandansky model [2], one can obtain estimates of the variance-covariance matrix of $(\hat{\mu}_0, \hat{\mu}_1, \ldots, \hat{\mu}_p)$. Since the

application involves a great deal of matrix algebra, the discussion of it appears in Appendix E. The matrix derived is $p + 1 \times p + 1$ and is labeled R. The estimated variance-covariance matrix is found by multiplying R by σ_A^2.

To obtain actual numbers for variances or covariances one must replace σ_A^2 by its estimate $\hat{\sigma}_A^2$. The latter is given by

$$\hat{\sigma}_A^2 = \frac{\sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2}{n - (p + 1)},$$

where

$$\hat{Y}_i = e^{\hat{\mu}_0} X_{1i}^{\hat{\mu}_1} \cdots X_{pi}^{\hat{\mu}_p} \quad \text{for } i = 1, 2, \ldots, n.$$

PREDICTION INTERVAL

As in the multiplicative case, a prediction interval for Y will be used to offer a measure of the "goodness" of the estimated regression function $\hat{Y} = e^{\hat{\mu}_0} X_1^{\hat{\mu}_1} \cdots X_p^{\hat{\mu}_p}$ as a predictor of Y. So take any new values of $(X_1, X_2, \ldots, X_p)$. Let

$$D = Y - \hat{Y}.$$

Then D is a random variable. For large n it is approximately normally distributed. The expected value and variance of D are given by

$$E\ D = E\ Y - E\ \hat{Y} = e^{\mu_p} X_1^{\mu_1} \cdots X_p^{\mu_p} - E\ \hat{Y}$$

and

$$\text{Var } D = \text{Var } Y + \text{Var } \hat{Y} = \sigma_A^2 + \text{Var } \hat{Y}.$$

Note there is no covariance term in Var D since Y is independent of $\hat{\mu}_0, \ldots, \hat{\mu}_p$ and hence of $\hat{Y}$.

The expected value and variance of $\hat{Y}$ can be approximately obtained from the first order Taylor series expansion of $\hat{Y}$ around the theoretical regression line. This is given by

$$\hat{Y} \approx e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}[1 + (\hat{\mu}_0 - \mu_0) + \ln X_1(\hat{\mu}_1 - \mu_1) + \cdots + \ln X_p(\hat{\mu}_p - \mu_p)].$$

Then

$$E\,\hat{Y} \approx e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}[1 + E\,(\hat{\mu}_0 - \mu_0) + \ln X_1\, E\,(\hat{\mu}_1 - \mu_1) + \cdots + \ln X_p\, E\,(\hat{\mu}_p - \mu_p)] \approx e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p},$$

since

$$E\,\hat{\mu}_k \approx \mu_k \quad \text{for} \quad k = 0, 1, \ldots, p.$$

Also,

$$\text{Var}\,\hat{Y} \approx \left(e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}\right)^2 \Big[\text{Var}\,\hat{\mu}_0 + 2\sum_{k=1}^{p} \text{Cov}\,(\hat{\mu}_0, \hat{\mu}_k)\ln X_k + \sum_{k=1}^{p} \text{Var}\,\hat{\mu}_k \ln^2 X_k + \sum_{k=1}^{p}\sum_{\substack{j=1 \\ j\neq k}}^{p} \text{Cov}\,(\hat{\mu}_k, \hat{\mu}_j)\ln X_k \ln X_j\Big]$$

$$= \sigma_A^2\left(e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}\right)^2 \left[(1, \ln X_1, \ldots, \ln X_p)\, R \begin{bmatrix} 1 \\ \ln X_1 \\ \vdots \\ \ln X_p \end{bmatrix}\right],$$

which can be approximated by

$$\sigma_A^2 \hat{Y}^2 W,$$

where

$$W = \left[(1,\ \ln X_1,\ \ldots,\ \ln X_p)\ R \begin{bmatrix} 1 \\ \ln X_1 \\ \vdots \\ \ln X_p \end{bmatrix}\right].$$

We therefore have that D is approximately normally distributed with

$$E\ D \approx 0$$

and

$$\text{Var } D \approx \sigma_A^2[1 + \hat{Y}^2 W].$$

Then

$$\frac{D}{\sqrt{\sigma_A^2[1 + \hat{Y}^2 W]}}$$

is (approximately) a standard normal random variable. A prediction interval can now be set up by estimating σ_A^2 by $\hat{\sigma}_A^2$. However, as in the multiplicative case, this would introduce an additional random variable whose effect can be controlled by the construction of an appropriate t-distributed random variable.

This is accomplished by noting again that

$$\frac{\hat{\sigma}_A^2[n - (p + 1)]}{\sigma_A^2}$$

has a chi-square distribution with $n - (p + 1)$ degrees of freedom and is independent of

$$\frac{D}{\sqrt{\sigma_A^2(1 + \hat{Y}^2 W)}}\ .$$

Hence

$$\frac{D}{\sqrt{\hat{\sigma}_A^2(1 + \hat{Y}^2 W)}}$$

has (approximately) a t-distribution with n - (p + 1) degrees of freedom.

Now to set up the prediction interval, choose the probability, γ, that is to be covered by the interval. Let

$$\alpha = \frac{1 + \gamma}{2}$$

and t_α be the $\alpha(100)$-th percentile of the t-distribution with n - (p + 1) degrees of freedom. As in the multiplicative case $t_{1-\alpha} = -t_\alpha$; hence

$$P\left\{-t_\alpha < \frac{D}{\sqrt{\hat{\sigma}_A^2(1 + \hat{Y}^2 W)}} < t_\alpha\right\} = \gamma$$

or

$$P\left\{\hat{Y} - t_\alpha \sqrt{\hat{\sigma}_A^2(1 + \hat{Y}^2 W)} < Y < \hat{Y} + t_\alpha \sqrt{\hat{\sigma}_A^2(1 + \hat{Y}^2 W)}\right\} = \gamma.$$

The latter is the prediction interval for Y. Thus a measure for the "goodness" of the predictive ability of Y has been provided for this model.

IV. COMPARISON OF THE MODELS

In the previous sections, problems 1 and 2 were discussed in relation to each of the two models considered. For each model, "best" estimators were defined, a method for finding them was given, and a prediction interval for Y was found.

Consider now the problem of deciding which model is best. If the analyst has some idea of how the error occurs in observing Y, i.e., is it additive or multiplicative, the choice appears easy. He chooses the model that corresponds to how the error occurs. But what if the analyst has no idea of how the error occurs? He then is faced with the problem of picking "best" estimators independent of what model is used. The choice of a criterion of what is best then affects the choice of models.

One way of handling this problem is to look at how the "best" estimators were obtained for the two models. For this discussion let $f_i(\mu_0, \ldots, \mu_p) = e^{\mu_0} x_{1i}^{\mu_1} \cdots x_{pi}^{\mu_p}$. In the additive model the estimators were picked to minimize

$$(1) \qquad \sum_{i=1}^{n} (Y_i - f_i)^2,$$

and in the multiplicative model the estimators were picked to minimize

$$(2) \qquad \sum_{i=1}^{n} (\ln Y_i - \ln f_i)^2.^{\dagger}$$

Now (2) is equivalent to

---

†In this case, μ_0 should be replaced by μ_0^* in order to explicitly include both cases. Expression (2) really only represents the median case.

$$(3) \qquad \sum_{i=1}^{n} \left[\ln \frac{Y_i}{f_i} \right]^2 .$$

So in the additive model one is interested in controlling the size of the actual deviations of Y_i from f_i, while in the multiplicative model one is essentially interested in controlling the relative deviations of Y_i from f_i. The analyst therefore may wish to choose the model (define "best") depending on what type of error he is most interested in controlling. For observations and predictions in a small interval of Y values (Y values of the same magnitude), the additive model might be preferred. For observations and predictions in a wider interval of Y values, the multiplicative model might be preferred.

Other criteria are available for the choice of "best," but their meaning as far as which model is more correct is hard to interpret. For instance, one might use the model that gives the narrowest prediction interval for the given values of $X_1, X_2, \ldots, X_p$. Another possibility is to choose the model that has the smaller estimated variance for Y. This is essentially the same as comparing the standard error of the estimates. The theoretical variance and the estimated variance are given in the table below.

Model	VAR Y		
	Theoretical	Estimated	
Additive	σ_A^2	$\hat{\sigma}_A^2$	(a)
Multiplicative			
Median Case	$Y^2 e^{\sigma_M^2}\left(e^{\sigma_M^2} - 1\right)$	$\hat{Y}^2 e^{\hat{\sigma}_M^2}\left(e^{\hat{\sigma}_M^2} - 1\right)$	(b)
Mean Case	$Y^2\left(e^{\sigma_M^2} - 1\right)$	$\hat{Y}^2\left(e^{\hat{\sigma}_M^2} - 1\right)$	(c)

In the table, $\hat{Y}$ is equal to the estimated regression function, and the estimated variances are given by

$$\hat{\sigma}_A^2 = \sum_{i=1}^{n} \frac{(Y_i - \hat{Y}_i)^2}{n - (p + 1)} \qquad \text{in (a)},$$

$$\hat{\sigma}_M^2 = \sum_{i=1}^{n} \frac{(\ln Y_i - \widehat{\ln Y_i})^2}{n - (p + 1)} \qquad \text{in (b) and (c)},$$

where the summations are over the historical data used for the solution of the estimators and $\widehat{\ln Y_i}$ is defined as in the multiplicative model, i.e., $\widehat{\ln Y_i} = \hat{\mu}_0^* + \hat{\mu}_1 \ln X_{1i} + \cdots + \hat{\mu}_p \ln X_{pi}$. Notice that (a) is independent of the values of $X_1, \ldots, X_p$ (except for the historical data), while (b) and (c) depend on the value of these variables through Y. So choice of the model with the smaller estimated variance will depend in general on the values of $X_1, \ldots, X_p$. Note that (b) and (c) are really the same, as the $\hat{Y}$ in (b) is equal to $e^{\hat{\sigma}_M^2}$ times the $\hat{Y}$ in (c).

It should be noted that only one possible criterion of "best" was examined in each model. Many other possible criteria are available, such as those parameters that minimize

$$\sum_{i=1}^{n} |Y_i - f_i|.$$

This definition might be appropriate for the additive model. Considering different criteria of "best" might lead to new comparisons for the two models.

It should be clear that the choice of which model is best is still up to the analyst, and it is hoped that this section has helped to clarify the issues involved in making that choice.

V. COMPUTER PROGRAM

INTRODUCTION

The computer program discussed in this section was developed explicitly to reflect the basic concepts presented in Sections I through IV of this Memorandum. Hence, a detailed description of the underlying principles of the program will not be given here, since such a discussion would essentially duplicate the preceding presentation. This section will consider mainly the program restrictions, the sequence of operations in the program, the input procedures for the program, and the outputs that result. However, for those readers who may be concerned with the operating details of the program, a listing of the FORTRAN-IV program is presented in Appendix F.

The reader will recall that the discussion in this Memorandum has been concerned with a regression function f consisting of p independent variables of the form

$$f = e^{\mu_0} \cdot X_1^{\mu_1} \cdot X_2^{\mu_2} \cdot \ldots \cdot X_p^{\mu_p},$$

where e = base of natural logarithms ≈ 2.71828. However, because of storage considerations, the regression function in the computer program is limited to a maximum of three independent variables:

$$f = e^{\mu_0} \cdot X_1^{\mu_1} \cdot X_2^{\mu_2} \cdot X_3^{\mu_3}.$$

The reason for this restriction is that the machine has a limitation of approximately 10^{38} for floating-point numbers. As was shown earlier, not only f, but f^2, must be calculated in the program. For more than

three independent variables, the latter could easily exceed 10^{38} in many cases. Hence, the decision was made to restrict the function to a maximum of three independent variables.

If three independent variables are entered in the input data, the program has the flexibility of considering (a) all three taken together.

$$Y = e^{\mu_0} \cdot X_1^{\mu_1} \cdot X_2^{\mu_2} \cdot X_3^{\mu_3},$$

or (b) all possible combinations (seven) of the independent variables taken in the following order:

$Y = e^{\mu_0} \cdot X_1^{\mu_1},$ First pass through program

$Y = e^{\mu_0} \cdot X_2^{\mu_2},$ Second pass through program

$Y = e^{\mu_0} \cdot X_3^{\mu_3},$

$Y = e^{\mu_0} \cdot X_1^{\mu_1} \cdot X_2^{\mu_2},$

$Y = e^{\mu_0} \cdot X_1^{\mu_1} \cdot X_3^{\mu_3},$

$Y = e^{\mu_0} \cdot X_2^{\mu_2} \cdot X_3^{\mu_3},$

$Y = e^{\mu_0} \cdot X_1^{\mu_1} \cdot X_2^{\mu_2} \cdot X_3^{\mu_3}.$ Seventh pass through the program

Similarly, if only two independent variables are entered, the program has the flexibility of considering either the two taken together,

$$Y = e^{\mu_0} \cdot X_1^{\mu_1} \cdot X_2^{\mu_2},$$

or all possible combinations (three) in the following order:

$$Y = e^{\mu_0} \cdot X_1^{\mu_1}, \qquad \text{First pass}$$

$$Y = e^{\mu_0} \cdot X_2^{\mu_2}, \qquad \text{Second pass}$$

$$Y = e^{\mu_0} \cdot X_1^{\mu_1} \cdot X_2^{\mu_2}. \qquad \text{Third pass}$$

Lastly, if only one independent variable is used, then, of course, only the one combination is considered by the program, i.e.,

$$Y = e^{\mu_0} \cdot X_1^{\mu_1}.$$

For any run in which more than one independent variable is entered in the input data, the program automatically considers all combinations of the independent variables, taken one combination at a time, unless a "flag" to the contrary is entered as an input on the title card. In that case, only the combination of all independent variables is considered.

With regard to notation of the variables and exponents, the following is used within the program:

$$Y = e^{A} \cdot W^{B} \cdot X^{C} \cdot Z^{D},$$

and in the mean case of the logarithmic form,

$$Y = e^{A1} \cdot W^{B} \cdot X^{C} \cdot Z^{D}.$$

However, since the W, X, Z notation might be somewhat confusing, or difficult to remember, the following has been adopted for output purposes:

$$Y = e^{A} \cdot X1^{B} \cdot X2^{C} \cdot X3^{D}.$$

In the above notation B represents the exponent of the first independent variable actually being used during a pass (regardless of whether it was entered as the first independent variable), C represents the exponent of the second independent variable actually being used, and D represents the exponent of the third independent variable. Suppose that only the second independent variable (X) of three is being considered during the second pass through the program. Then, internally the X variables are changed to W variables, and the program uses the relationship

$$Y = e^{A} \cdot W^{B},$$

even though in this case W actually represents the second independent variable entered on the input cards.

The method by which the independent variables are manipulated in this way within the program is as follows. The values of the first independent variable are entered as $W1_i$, the second as $X1_i$, and the third as $Z1_i$, where i represents each point. The W_i, X_i, and Z_i values used in the computations are generated during each pass from the input values ($W1_i$, $X1_i$, $Z1_i$), depending on which combination of independent variables is being considered for that pass. For example, suppose that Y is being considered as a function of the second and third independent variables (entered as $X1_i$ and $Z1_i$). In this case, the machine performs the following operations within the program for each data point:

$$W_i = X1_i,$$
$$X_i = Z1_i,$$
$$Z_i = 1,$$

it then uses the following relationship for that pass through the program:

$$Y_i = e^A \cdot W_i^B \cdot X_i^C \cdot Z_i^D,$$

or, actually,

$$Y_i = e^A \cdot W_i^B \cdot X_i^C$$

(since all $Z_i = 1.0$).

Table 1 summarizes the operations for three independent variables taken in all combinations.

RESTRICTIONS

As indicated previously, the computer program makes use of several auxiliary subroutines, one of which is the matrix inversion subroutine. For the matrix inversion subroutine, the time required to invert an n × n matrix is approximately proportional to n^3. Therefore it can be seen that the time required for inversion increases very rapidly as the size of the matrix is increased. For a 50 × 50 matrix, the time required for inversion is about 1-1/3 minutes, and for a 75 × 75 matrix, the time required is about 4-1/2 minutes. Since the program requires the inversion of an $[N - (p + 1)] \times [N - (p + 1)]$ matrix, where N is the number of data points and p is the number of independent variables, the decision was made to limit the maximum allowable number of data points in one set of data to 50 for the program. (A similar situation also exists for the eigenvector and eigenvalue routine.)

The minimum number of data points entered must be two more than the maximum number of independent variables being used. This

Table 1

SUMMARY OF OPERATIONS FOR THREE SETS OF INDEPENDENT VARIABLES TAKEN IN ALL COMBINATIONS

Input	Pass through Program	Internal Operations	Output
Y_i, $W1_i$, $X1_i$, $Z1_i$	1	First independent variable: $W_i = W1_i$ $X_i = 1.$ $Z_i = 1.$ $Y_i = e^A \cdot W_i^B \cdot X_i^C \cdot Z_i^D$	$Y_i = e^A \cdot X1_i^B$
	2	Second independent variable: $W_i = X1_i$ $X_i = 1.$ $Z_i = 1.$ $Y_i = e^A \cdot W_i^B \cdot X_i^C \cdot Z_i^D$	$Y_i = e^A \cdot X2_i^B$
	3	Third independent variable: $W_i = Z1_i$ $X_i = 1.$ $Z_i = 1.$ $Y_i = e^A \cdot W_i^B \cdot X_i^C \cdot Z_i^D$	$Y_i = e^A \cdot X3_i^B$

For note, see page 45.

[continued]

Table 1 -- continued

Input	Pass through Program	Internal Operations	Output
Y_i, $W1_i$, $X1_i$, $Z1_i$	4	First and second independent variables: $W_i = W1_i$ $X_i = X1_i$ $Z_i = 1.$ $Y_i = e^A \cdot W_i^B \cdot X_i^C \cdot Z_i^D$	$Y_i = e^A \cdot X1_i^B \cdot X2_i^C$
	5	First and third independent variables: $W_i = W1_i$ $X_i = Z1_i$ $Z_i = 1.$ $Y_i = e^A \cdot W_i^B \cdot X_i^C \cdot Z_i^D$	$Y_i = e^A \cdot X1_i^B \cdot X3_i^C$
	6	Second and third independent variables: $W_i = X1_i$ $X_i = Z1_i$ $Z_i = 1.$ $Y_i = e^A \cdot W_i^B \cdot X_i^C \cdot Z_i^D$	$Y_i = e^A \cdot X2_i^B \cdot X3_i^C$

For note, see page 45.

[continued]

Table 1 -- continued

Input	Pass through Program	Internal Operations	Output
Y_i, $W1_i$, $X1_i$, $Z1_i$	7	All three independent variables: $W_i = W1_i$ $X_i = X1_i$ $Z_i = Z1_i$ $Y_i = e^A \cdot W_i^B \cdot X_i^C \cdot Z_i^D$	$Y_i = e^A \cdot X1_i^B \cdot X2_i^C \cdot X3_i^D$

NOTE: $i = 1, \ldots, N$, $N \leq 50$.

is because the degrees of freedom used in some of the statistical calculations are

N - (number of parameters),

or

N - (number of independent variables + 1).

Therefore, in order for the degrees of freedom to be greater than zero, N must be at least two greater than the number of sets of independent variables being used.

The program is also restricted to values greater than zero for all dependent and independent variable data points being considered, since the logarithm of each such point is calculated. If any of these data points were to have a zero or negative value, its logarithm would be meaningless (i.e., could not be calculated), and the run would stop.

SEQUENCE OF OPERATIONS

As many runs as are desired, or as time permits, may be made by the program. The machine reads in one set of data at a time, performs all of the calculations and resulting printouts on that set of data (for up to seven passes through the program), and then reads in a new set of data. When there are no more sets of data to be read in, the run is terminated automatically.

The program consists principally of a main routine, which in turn calls most of the subroutines as required. The program is so structured that the logarithmic model (multiplicative error assumed)

is treated first.† The solutions that are obtained for the parameters (A, B, C, D) in this case are then used as the starting points (initial guesses) for the exponential model (additive error assumed). It may be recalled that the logarithmic model gives exact solutions, whereas the exponential model is based on an iterative, correction-type operation.‡ In this operation, initial guesses are made for the parameters, and then corrections, guaranteed to produce convergence to a solution, are made to these initial guesses. This process is repeated over and over again until the change in the value of each parameter from one iteration to the next becomes equal to or less than some predetermined value (10^{-8} in the program). At that point a solution is assumed.††

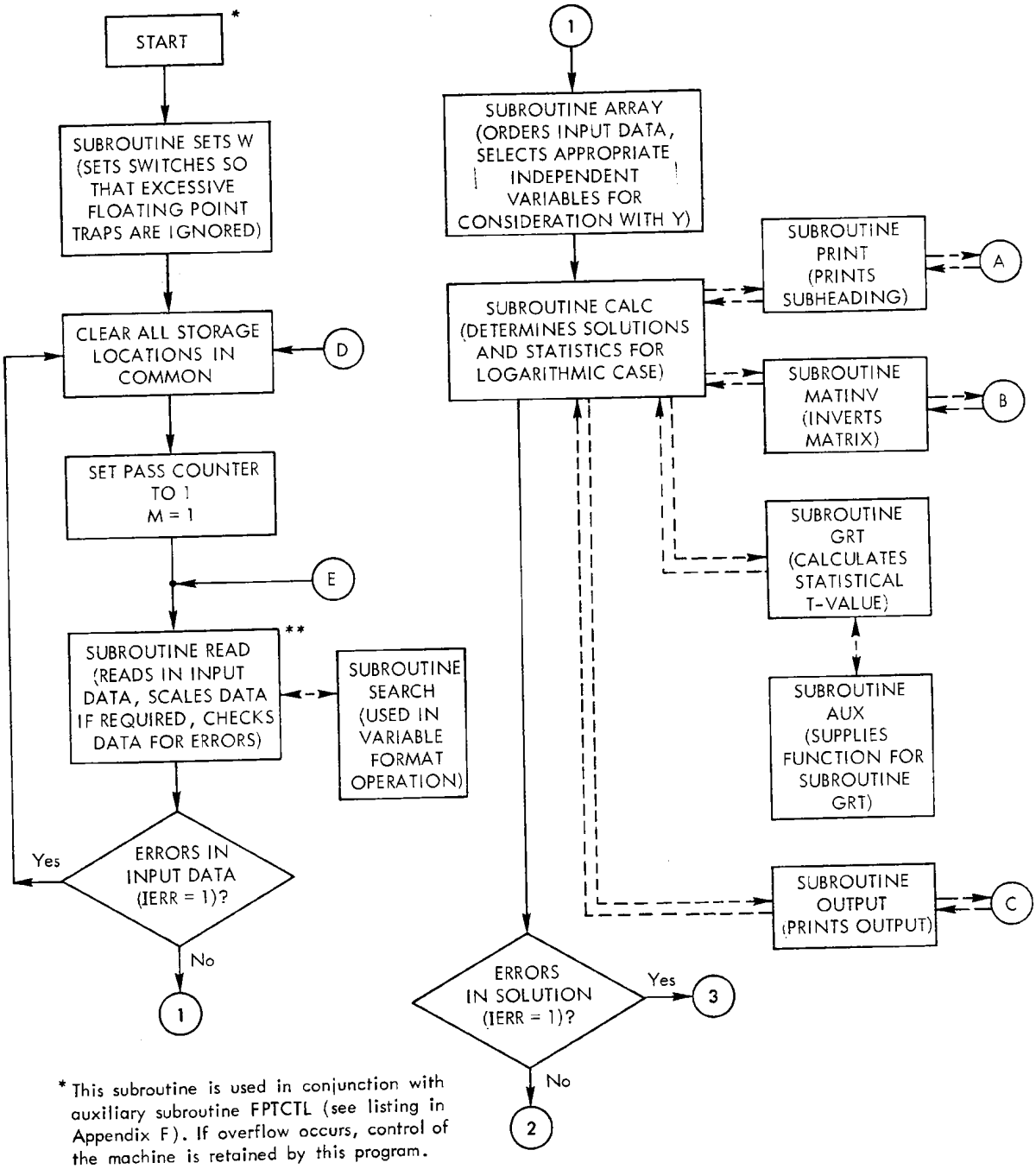
The general flow of operations is depicted in Fig. 1. At the beginning of the main program, an auxiliary subroutine SETSW (RAND W037) is called on to set the four control switches and trap limit counter. This is done so that control of the machine will be retained by the program if excessive floating-point traps are encountered. Next, all of the variables listed in common are cleared (set equal to zero). The program then calls on subroutine READ for reading in one set of data. In subroutine ARRAY the data are ordered with respect to the dependent variable (Y_i) and are then transformed as required, depending on which independent variables are being considered during that pass through the program.

---

†Throughout this section and also in Appendix F, the expressions "model" and "case" are used somewhat interchangeably. It may also be noted that the expression "logarithmic model" refers to the model in which a multiplicative error is assumed (Sec. II), and "exponential model" refers to the model in which an additive error is assumed (Sec. III).

‡See Sec. III.

††See [3].



* This subroutine is used in conjunction with auxiliary subroutine FPTCTL (see listing in Appendix F). If overflow occurs, control of the machine is retained by this program. Underflows are ignored.

** If there are no more sets of input data to be read, run is terminated automatically at this point.

Fig. 1—Flow of operations

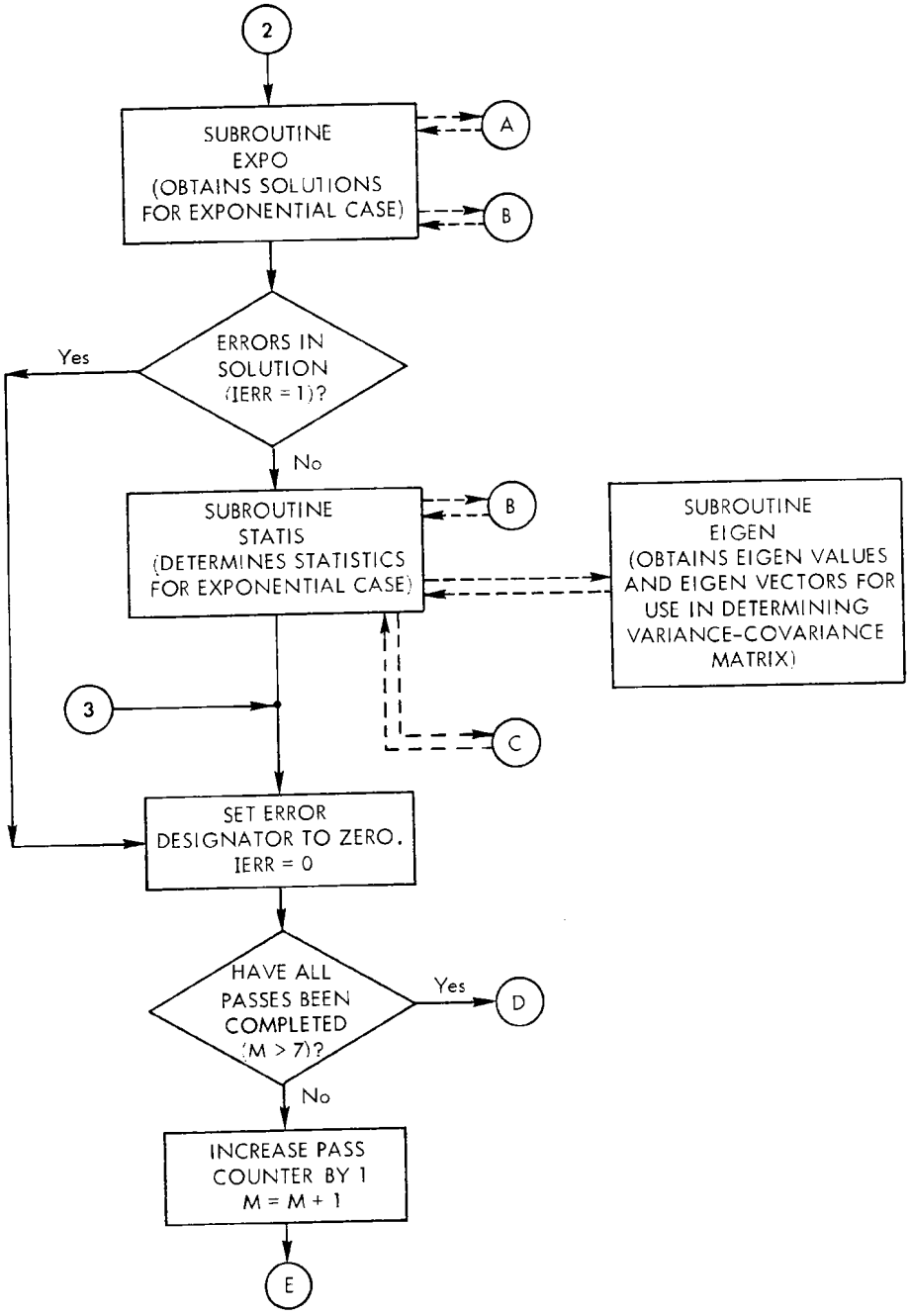


Fig. 1—Flow of operations (cont'd)

Subroutine CALC is next called on to obtain solutions for the parameters (A, B, C, D) in the logarithmic model. However, it first calls on subroutine PRINT to print a heading stating which independent variables are being considered. After solutions are obtained for the parameters, subroutine CALC then initiates the determination of the Student's statistical t-value by calling on the auxiliary, general-root-finder (GRT) subroutine (RAND W008).† Following this are all of the statistical calculations relating to the logarithmic model. Lastly in this subroutine, another subroutine, OUTPUT, is called on to print, according to prescribed formats, all of the pertinent data relating to the logarithmic model.

Using the solutions from the logarithmic model as the starting points, subroutine EXPO is called on to obtain least-squares solutions of the parameters for the exponential model. If for some reason no solutions are obtained within the 50 iterations allowed, a message to this effect is printed, and the machine continues on to the next pass.

Subroutine STATIS calculates the statistical measures required for the exponential model and then calls on subroutine OUTPUT to print the pertinent data relating to the exponential model.

The above sequence of operations may be repeated using another combination of independent variables for the same set of data. When all of the operations relating to the one set of data have been completed, another set of data is read in, and the sequence of operations starts over again.

---

†The determination of the t-value is based on an approximation method described in [6]. (It should be noted that the GRT subroutine also calls on subroutine AUX to supply the approximation equation.)

Tables 2 and 3 summarize the functional operations of the program subroutines and the auxiliary subroutines required for the program.

INPUT PROCEDURES

The input cards consist of the following in the order in which they must be entered:

1. Title card
2. Order card
3. Format card

} Need only to be entered first time if data for all runs are to be read in same format

4. Data cards
5. Blank card

Title Card

The title card, which must always be entered for each set of data, performs two basic functions. It gives some initial instructions to the machine, and it contains the title for the run. If the variable format cards (the next two cards) are to be read in for this run, a "1" is entered in Col. 1; otherwise this column is left blank.

Column 2 is used to indicate whether all possible combinations of the independent variables are to be considered (Col. 2 left blank) or whether all of the independent variables are to be considered together ("1" in Col. 2). For example, suppose that data for three independent variables are entered. Then, a "1" in Col. 2 will cause the machine to consider only the following relationship:

$$Y = e^{A} \cdot X1^{B} \cdot X2^{C} \cdot X3^{D}.$$

Table 2

SUMMARY OF SUBROUTINES IN COMPUTER PROGRAM

Subroutines	Functions
MAIN ROUTINE	Controls entire program.
READ	Reads in data.
ARRAY	Orders data and selects independent variable(s) to be used for each pass.
CALC (logarithmic model) (multiplicative error assumed)	Obtains solutions of parameters A, B, C, D and calculates associated statistics. Calls subroutine OUTPUT for printouts.
PRINT	Prints headings stating which independent variables are being used.
AUX	Supplies approximation equation to general root finder auxiliary subroutine for determining statistical t-value.
EXPO (exponential model) (additive error assumed)	Obtains solutions of parameters A, B, C, D by iterative method.
STATIS	Calculates statistics for exponential case. Calls subroutine OUTPUT for printouts.
OUTPUT	Prints out results for each case.

Table 3

SUMMARY OF AUXILIARY SUBROUTINES

Subroutines	Functions
General root finder (GRT) (RAND W008)	Determines the Student's statistical t-value from the approximation equation supplied by subroutine AUX.
Matrix inversion with accompanying solution of linear equation (MATINV) (modified RAND W019)	Solves the matrix equation AX = B, where A is a square matrix and B is a matrix of constant vectors; A^{-1} is also obtained.
Eigenvalues and eigenvectors of a real symmetric matrix (EIGEN) (modified RAND W021)	Determines all scalar solutions λ_i (including proper multiplicity), and optionally the associated unit vectors, X_i, to the matrix equation $AX = \lambda X$, where A is a real, symmetric matrix.
FPT utility (SETSW) (RAND W037)	Provides a means of setting the four control switches and the trap limit counter in FPT. For this program, it is used to prevent stoppage of the run if excessive floating-point traps are encountered.
Image and Search (SEARCH) (RAND W062)	Image--picks up a word from a Hollerith list and stores it in a specified location. Search--matches a word with the words in a Hollerith list and stores the number of the word from the list. Generally used in a computed GO TO statement. W062 is used in reading in the order of the data from the order card.
FPTCTL	In conjunction with SETSW, provides a way of retaining control of the machine if overflows occur.

If Col. 2 is left blank, all seven possible combinations of the independent variables, taken one at a time, will be considered for this set of data.

If data for only one independent variable are entered, Col. 2 may be left blank.

Column 3 is used to indicate the level of significance to be used in the program. Four choices are available as follows:

Value in Col. 3	Level of Significance
1	0.01
2	0.05
3	0.1
4	0.2

If Col. 3 is left blank, a level of significance of 0.05 is used automatically.

Column 4 is not used. The next four sets of two columns each are used for scaling the input dependent and independent variable data when required. This is done as follows:

Variable Scaled	Column Location of Scale Designator
Y (dependent variable)	5 - 6
X1 (first independent variable)	7 - 8
X2 (second independent variable)	9 - 10
X3 (third independent variable)	11 - 12

A fixed-point number is used to indicate the number of places that the decimal is to be moved. A positive number indicates that the decimal is to be moved to the right, and a negative number indicates that the decimal is to be moved to the left. For example, suppose that a "3" is

entered in Col. 6. Then each input Y value will have its decimal point moved three places to the right, e. g., 50.123 → 50123., 0.6217 → 621.7, etc. Care must be taken to enter all positive factors in the right-hand column of the two-column set. Thus, in the above example, if the "3" were entered in Col. 5 instead of Col. 6, the machine would read this number as "30" instead of "3." It should also be noted that the scale factor so entered applies to the entire set of the corresponding variable.†

Columns 13-14 are also not used. Columns 15-72 are reserved for the title or other identifying remarks. Any keypunch symbols (alphanumeric) may be entered in these columns.

An example of a title card is shown in Fig. 2, and a summary of the above information is given in Table 4.

Order Card

The next card (when used as indicated by a "1" in Col. 1 of the title card) indicates the order (from left to right) in which the data sets are located on the data cards. The order (alphanumeric) is entered in Cols. 1-2, 4-5, 7-8, 10-11, and 13-14 (Cols. 7-14 may not be required). The symbols used to show the order are as follows:

Symbol	Type of Data
ID	Identifier (alphanumeric) (if used)
Y1	Dependent variable (required)
X1	First independent variable (required)
X2	Second independent variable (if used)
X3	Third independent variable (if used)

†Preferably, the range of values for each variable should be kept approximately the same and reasonably low so that excessive overflows are not encountered. Scaling does not affect the values of the exponents of the independent variables.

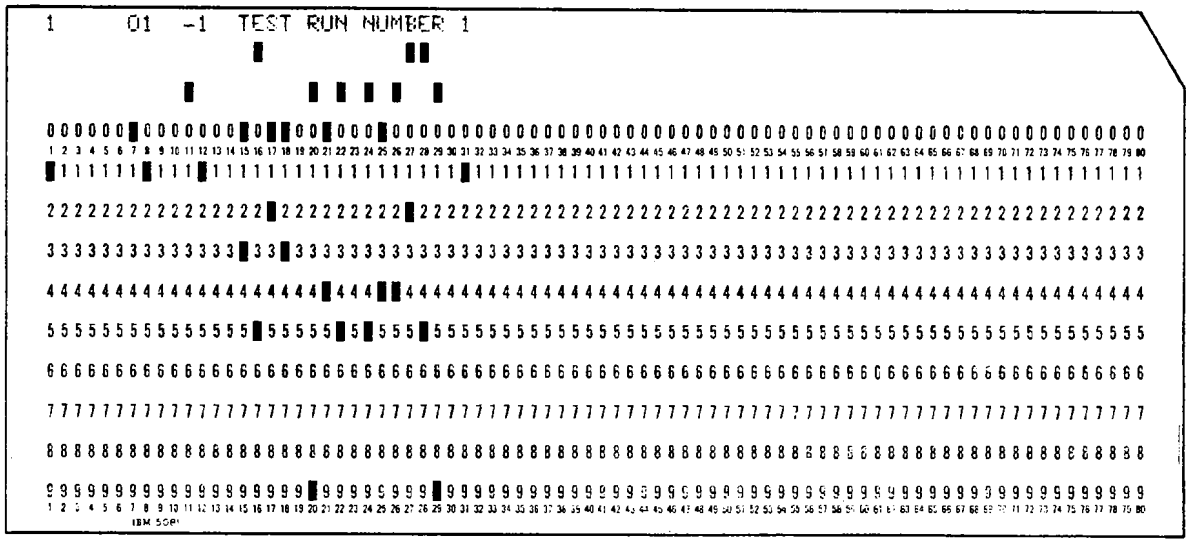


Fig. 2—Example of title card

Table 4

SUMMARY OF INFORMATION ON TITLE CARD

Columns	Remarks
1	A "1" indicates that the order card and format card are to be read following this card. If blank, no such cards are to be read.
2	A "1" indicates that all independent variables taken together are to be considered. If blank, all combinations taken one at a time are to be considered.
3	Level of significance (LOS) designator. "1"--LOS of 0.01; "2"--LOS of 0.05; "3"--LOS of 0.1; "4"--LOS of 0.2. If blank, LOS of 0.05 is used automatically.
4	Not used.
5-6	Scale factor for dependent variable (Y). If positive, must be entered in Col. 6.
7-8	Scale factor for first independent variable (X1). If positive, must be entered in Col. 8.
9-10	Scale factor for second independent variable (X2). If positive, must be entered in Col. 10.
11-12	Scale factor for third independent variable (X3). If positive, must be entered in Col. 12.
13-14	Not used.
15-72	Title for run. May consist of any keypunch symbols (alphanumeric).

Suppose that a set of data is to be entered, in which the identifier, say, the date, is in Cols. 1-6; the dependent variable is in Cols. 10-19; and three independent variables are in Cols. 20-29, 30-39, and 40-49, respectively. Then, "ID" would be entered in Cols. 1-2 (since the identifier would be the first item of data to be read on the card); "Y1" would be entered in Cols. 4-5, and "X1," "X2," and "X3" would be entered, respectively, in Cols. 7-8, 10-11, and 13-14. This is shown in Fig. 3.

Format Card

The format card indicates where the data are located on the data cards. Again, this card is used only if a "1" is entered in Col. 1 on the title card. This card must begin with a left-hand parenthesis in Col. 1 and end with a right-hand parenthesis before Col. 73. For the above example, the format card would read as shown in Fig. 4.

Data Cards

The data cards (≤ 50) contain the input data relating to the dependent and independent variables and one identifier (if used) for each data point. The location of the data on these cards must be in exact agreement with the information entered on the format cards, or else they will not be read in properly. The numerical data (dependent and independent variables) must be entered as floating-point numbers, and the identifiers (if used) are entered as alphanumeric data. As stated previously, any set of numerical data being used must <u>not</u> contain any zeros or negative values. An example of a data card, for the above example, containing an identifier, a dependent variable, and the three corresponding independent variables is shown in Fig. 5.

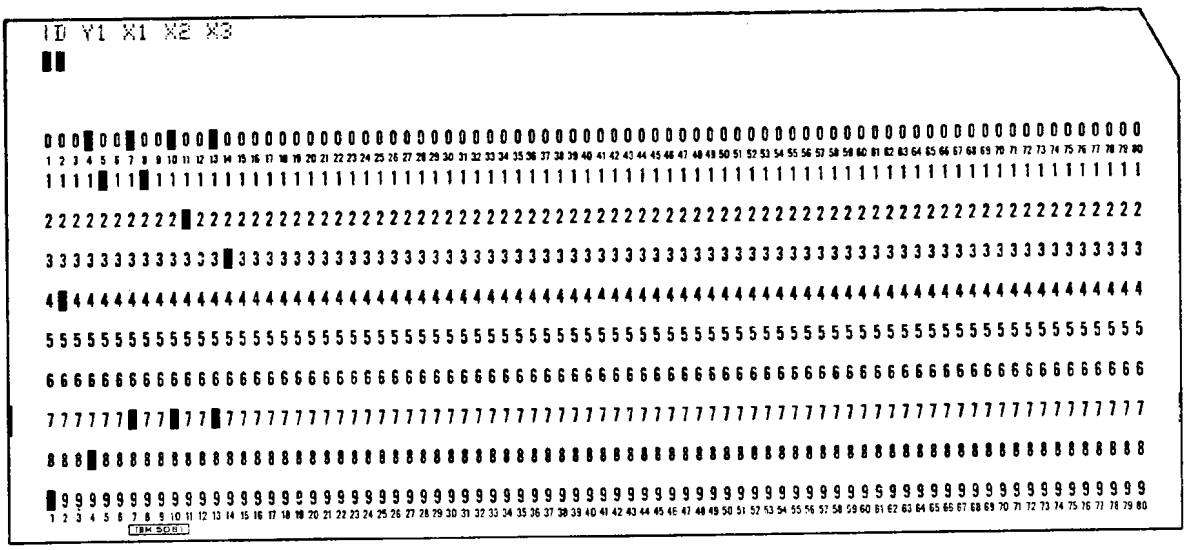


Fig.3—Example of order card

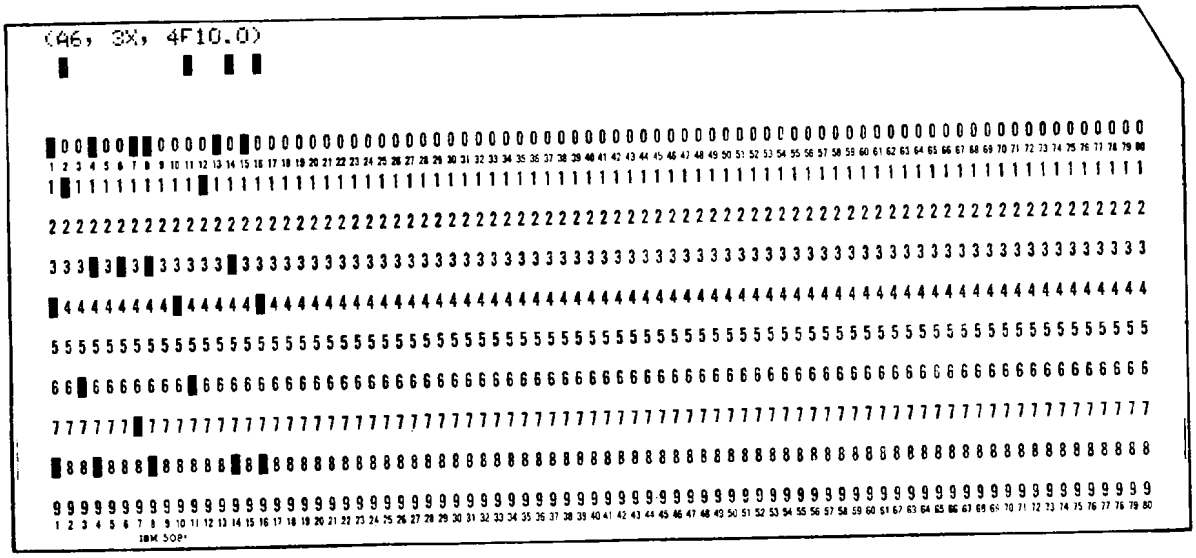


Fig.4—Example of format card

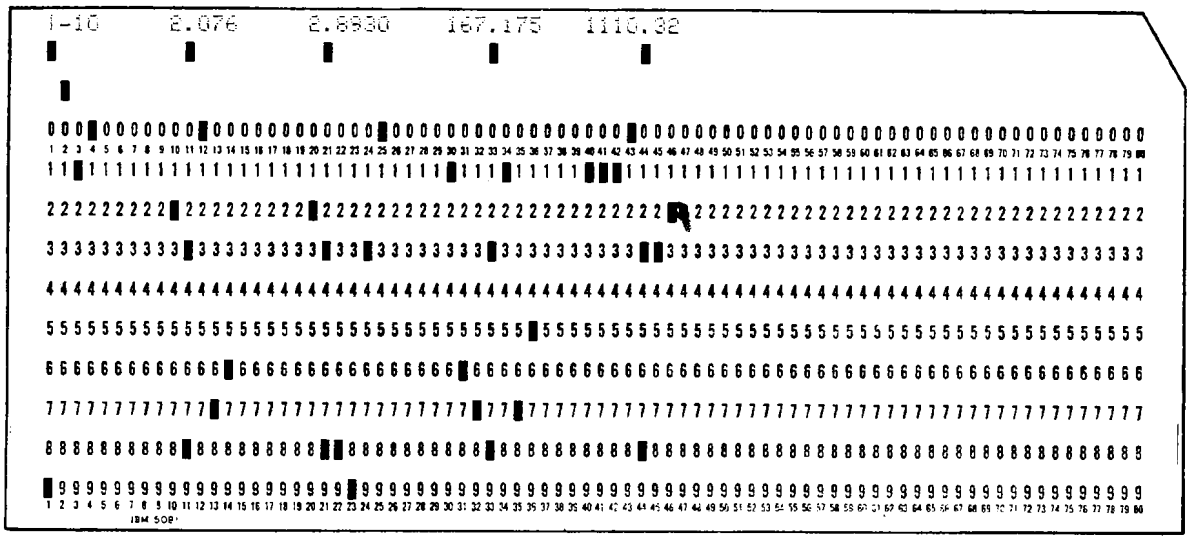


Fig.5—Example of data card

Blank Card

A blank card is used to signify the end of each data set. Consequently, this card must always be present at the end of each set of data cards.

Summary of Input Cards

Figure 6 shows the order of data cards for two runs in which the second set is to be read in the same format as the first set.

OUTPUTS

Examples of the outputs that result from this program are shown in Figs. 7-11. The particular data shown have no meaning whatsoever, and are presented only for the purpose of displaying the outputs. Since most of the headings are quite self-explanatory, no detailed explanation will be given.

Figures 7-9 show the outputs relating to the logarithmic case. The scale factors (if used) are printed beneath the main title. For this problem the first set of independent variables entered (X1) were scaled up by a factor of 10 (i.e., 10^1). The second set were scaled down by a factor of 10 (i.e., 10^{-1}). Thus, "Base 10 Exponent 1," for example, is read as 10^1, or 10. The scale factors are printed only on the first page of the output (logarithmic case), but, of course, they apply to the data for the exponential case as well.

The values under the last two headings on the third line:

EXP(A)	EXP(A)
(MEDIAN)	(MEAN)

are, respectively, e^A for the median and mean cases. (Values for A are

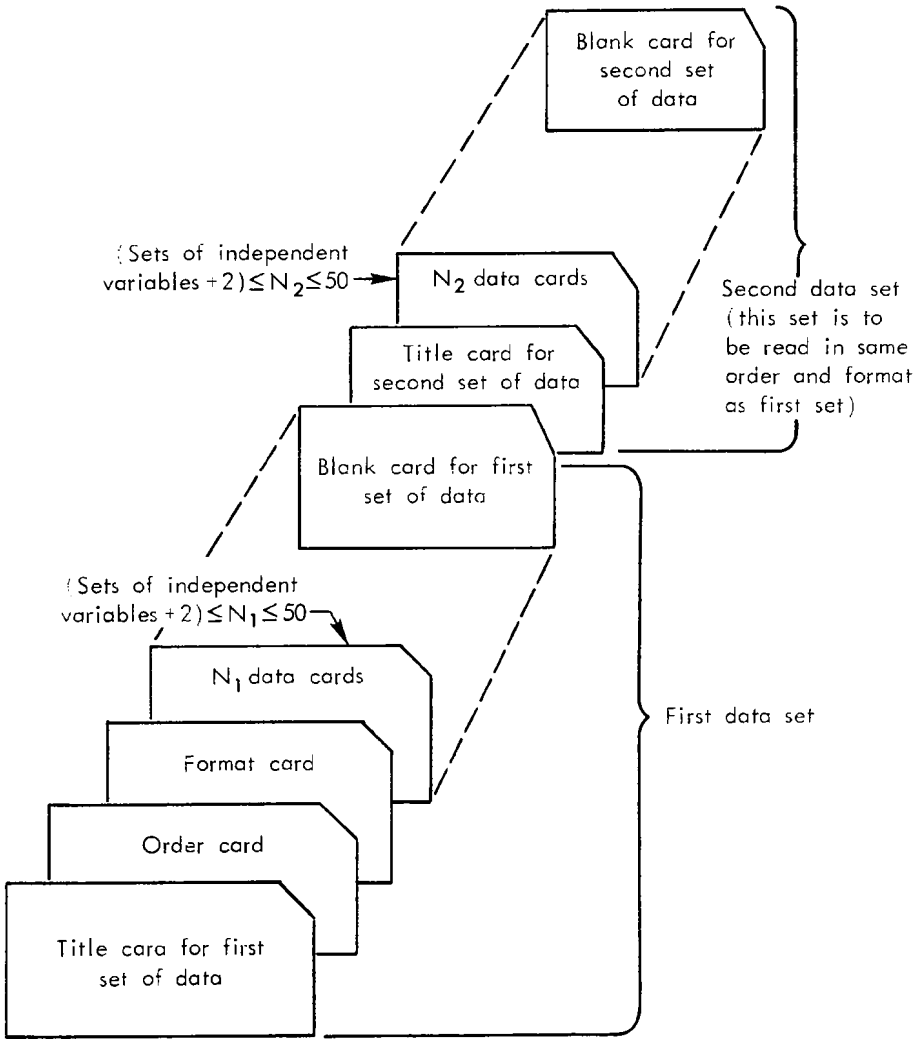


Fig.6—Arrangement of data cards for two runs

TEST RUN NUMBER 1

X1 SCALE FACTOR = BASE 10 EXPONENT 1

X3 SCALE FACTOR = BASE 10 EXPONENT -1

LOGARITHMIC CASE
(MULTIPLICATIVE ERROR ASSUMED)

Y AS A FUNCTION OF X1, X2, AND X3

A(MEDIAN)	A(MEAN)	B	C	D	Y MEAN
0.07130	0.07136	0.39558	1.91355	-2.22213	1.89059

SUM OF SQ. OF YDEV (MEDIAN)	SUM OF SQ. OF YDEV (MEAN)	STD. DEV. OF INPUT Y (ADJUSTED)	STD. DEV. OF A(MEDIAN)	STD. DEV. OF A(MEAN)	STD. DEV. OF B	STD. DEV. OF C	STD. DEV. OF D	COV. OF A,B
0.00476	0.00476	0.35221	0.51374	0.51374	0.14004	0.86371	0.73353	0.07177

COV. OF A,C	COV. OF A,D	COV. OF B,C	COV. OF B,D	COV. OF C,D	EXP(A) (MEDIAN)	EXP(A) (MEAN)
-0.43959	0.37057	-0.11868	0.09977	-0.63287	1.07390	1.07397

T VALUE = 2.20162

A (MEDIAN CASE) IS NOT SIGNIFICANT

A (MEAN CASE) IS NOT SIGNIFICANT

B IS SIGNIFICANT

C IS SIGNIFICANT

D IS SIGNIFICANT

Fig. 7—First page of output (logarithmic model)

. . . . . INPUT DATA					. . . . . CALCULATED DATA FOR MEDIAN CASE					
LABEL	Y	X1	X2	X3	Y CALC.	YDEV (Y-YCALC.)	STD. ERROR OF ESTIMATE (ADJUSTED)	COEFF. OF VARIATION (PERCENT)	LOWER PREDICT. LIMIT (LEVEL OF SIGNIF. = 0.05)	UPPER PREDICT. LIMIT
I-1	0.880	1.090	13.000	10.100	0.882	-0.003	0.00989	0.52294	0.852	0.913
I-2	1.436	4.150	30.291	21.318	1.437	-0.001	0.01610	0.85151	1.397	1.478
I-3	1.639	7.250	47.000	32.510	1.626	0.013	0.01822	0.96354	1.575	1.678
I-4	1.801	10.000	64.510	43.690	1.755	0.046	0.01966	1.04001	1.704	1.807
I-5	1.842	13.420	81.620	54.889	1.862	-0.020	0.02087	1.10374	1.814	1.912
I-6	1.900	16.450	99.020	66.123	1.932	-0.032	0.02164	1.14482	1.880	1.985
I-7	1.960	19.720	114.800	76.830	1.973	-0.013	0.02211	1.16935	1.921	2.026
I-8	2.000	22.610	133.000	88.500	2.016	-0.016	0.02259	1.19471	1.964	2.068
I-9	2.042	25.650	150.055	99.580	2.054	-0.012	0.02301	1.21721	2.001	2.108
I-10	2.076	28.930	167.175	111.032	2.079	-0.003	0.02330	1.23245	2.026	2.134
I-11	2.110	32.000	184.000	122.170	2.102	0.008	0.02356	1.24605	2.048	2.159
I-12	2.125	34.550	201.225	133.200	2.122	0.003	0.02378	1.25794	2.067	2.179
I-13	2.155	38.100	218.333	144.463	2.153	0.002	0.02413	1.27625	2.097	2.211
I-14	2.183	41.000	235.500	155.675	2.170	0.013	0.02432	1.28619	2.113	2.229
I-15	2.210	44.250	252.731	166.842	2.195	0.015	0.02459	1.30089	2.136	2.255

Fig.8—Second page of output (logarithmic model, median case)

INPUT DATA					CALCULATED DATA FOR MEAN CASE					
LABEL	Y	X1	X2	X3	Y CALC.	YDEV (Y-YCALC.)	STD. ERROR OF ESTIMATE (ADJUSTED)	COEFF. OF VARIATION (PERCENT)	LOWER PREDICT. LIMIT (LEVEL OF SIGNIF. = 0.05)	UPPER PREDICT. LIMIT
I-1	0.880	1.090	13.000	10.100	0.882	-0.003	0.00989	0.52294	0.852	0.913
I-2	1.436	4.150	30.291	21.318	1.437	-0.001	0.01610	0.85151	1.397	1.478
I-3	1.639	7.250	47.000	32.510	1.626	0.013	0.01822	0.96354	1.575	1.678
I-4	1.801	10.000	64.510	43.690	1.755	0.046	0.01966	1.04001	1.704	1.807
I-5	1.842	13.420	81.620	54.889	1.862	-0.020	0.02087	1.10374	1.814	1.912
I-6	1.900	16.450	99.020	66.123	1.932	-0.032	0.02164	1.14482	1.880	1.985
I-7	1.960	19.720	114.800	76.830	1.973	-0.013	0.02211	1.16935	1.921	2.026
I-8	2.000	22.610	133.000	88.500	2.016	-0.016	0.02259	1.19471	1.964	2.068
I-9	2.042	25.650	150.055	99.580	2.054	-0.012	0.02301	1.21721	2.001	2.108
I-10	2.076	28.930	167.175	111.032	2.080	-0.004	0.02330	1.23245	2.026	2.134
I-11	2.110	32.000	184.000	122.170	2.103	0.007	0.02356	1.24605	2.048	2.159
I-12	2.125	34.550	201.225	133.200	2.123	0.002	0.02378	1.25794	2.067	2.179
I-13	2.155	38.100	218.333	144.463	2.153	0.002	0.02413	1.27625	2.097	2.211
I-14	2.183	41.000	235.500	155.675	2.170	0.013	0.02432	1.28619	2.113	2.229
I-15	2.210	44.250	252.731	166.842	2.195	0.015	0.02459	1.30089	2.136	2.255

Fig.9—Third page of output (logarithmic model, mean case)

TEST RUN NUMBER 1

EXPONENTIAL CASE
(ADDITIVE ERROR ASSUMED)

Y AS A FUNCTION OF X1, X2, AND X3

A	B	C	D	AVG. PCT. DEV. OF Y	Y MEAN	STD. ERROR OF ESTIMATE (ADJUSTED)	COEFF. OF VARIATION (PERCENT)	COEFF. OF CORRELATION (UNADJUSTED)
0.06486	0.38496	1.83868	-2.13179	0.71376	1.89059	0.02050	1.08444	0.99867

SUM OF SQ. OF YDEV	STD. DEV. OF INPUT Y (ADJUSTED)	STD. DEV. OF A	STD. DEV. OF B	STD. DEV. OF C	STD. DEV. OF D	COV. OF A,B	COV. OF A,C	COV. OF A,D
0.00462	0.35221	0.51515	0.14331	0.85053	0.72056	0.07326	-0.43118	0.36007

COV. OF B,C	COV. OF B,D	COV. OF C,D	EXP(A)
-0.11676	0.09671	-0.61147	1.06701

T VALUE = 2.20162

A IS NOT SIGNIFICANT

B IS SIGNIFICANT

C IS NOT SIGNIFICANT

D IS SIGNIFICANT

Fig. 10—Fourth page of output (exponential model)

. . . INPUT DATA . . .		. . . . . CALCULATED DATA					
LABEL	Y	Y CALC.	YDEV (Y-YCALC.)	YDEV/Y (PERCENT)	2YCALC. - Y	LOWER PREDICT. LIMIT (LEVEL OF SIGNIF. = 0.05)	UPPER PREDICT. LIMIT
I-1	0.87980	0.89075	-0.01095	-1.24474	0.90170	0.82871	0.95279
I-2	1.43600	1.43593	0.00007	0.00508	1.43585	1.38540	1.48646
I-3	1.63900	1.62371	0.01529	0.93272	1.60843	1.56762	1.67981
I-4	1.80100	1.75184	0.04916	2.72980	1.70267	1.69906	1.80461
I-5	1.84200	1.85896	-0.01696	-0.92076	1.87592	1.81027	1.90765
I-6	1.90000	1.92851	-0.02851	-1.50030	1.95701	1.87840	1.97862
I-7	1.96000	1.97093	-0.01093	-0.55788	1.98187	1.92119	2.02068
I-8	2.00000	2.01429	-0.01429	-0.71462	2.02858	1.96710	2.06149
I-9	2.04200	2.05281	-0.01081	-0.52934	2.06362	2.00532	2.10030
I-10	2.07600	2.07949	-0.00349	-0.16808	2.08298	2.03239	2.12659
I-11	2.11000	2.10324	0.00676	0.32039	2.09648	2.05527	2.15121
I-12	2.12500	2.12392	0.00108	0.05062	2.12285	2.07562	2.17223
I-13	2.15500	2.15525	-0.00025	-0.01172	2.15551	2.10676	2.20375
I-14	2.18300	2.17270	0.01030	0.47176	2.16240	2.12358	2.22183
I-15	2.21000	2.19788	0.01212	0.54859	2.18575	2.14818	2.24758

Fig. 11—Fifth page of output (exponential model)

given on the top line.) Each e^A value thus represents the coefficient or the intercept of the regression function (point at which values of independent variables are 1).

The input data listed in Fig. 8 represent the actual data that are being used for that pass through the program. If more than one pass is to occur, then this listing will change each time with regard to the independent variable. For this problem, this output represents the seventh pass through the program.

For the convenience of the reader, the calculated data shown opposite the input data are listed for both the median and mean cases of the logarithmic model (Figs. 8 and 9). The only differences in these listings are the calculated Y values and the residual values ($Y_{input} - Y_{calc}$).

As explained in Sec. IV on the comparison of the models, values of the standard error of the estimate (and hence the coefficient of variation) may be used to compare the additive and exponential models.† For example, a comparison of these values in Fig. 8 with the standard error of the estimate and the coefficient of variation in the exponential model (Fig. 10) shows that for most of the points, the exponential model gives lower values for this particular run.

Figures 10 and 11 show the outputs for the exponential case. In Fig. 10, a "COEFF. OF CORRELATION (UNADJUSTED)" value is listed at the end of the top line. This value is not calculated for the logarithmic case, since it would have no real meaning.

In Fig. 11 there is a heading labeled "2YCALC. - Y," the significance

†It may be noted that as explained in Sec. IV, the variance of Y_{calc} in the logarithmic case varies with each point, whereas it is constant over the range of points in the exponential case.

of which is explained in Appendix D. If all of the values under this heading are positive for a given set of data, then the solutions of the parameters for the exponential case represent solutions that give an absolute minimum for the sum of the squares of the Y deviations in the region of (A, B, C, D) defined by $2Y_{i_{calc}} - Y_i > 0$. If the values are not all positive, a message noting this fact is printed.

The identifiers are listed in the left-hand column under the heading "LABEL." As alphanumeric input data they are used to identify each data point when so desired. However, as stated on page 58, they may be omitted in the input data.

The only other major difference in the output formats for the two cases is that a percent Y deviation is not given for the logarithmic case. Because of space limitations, it was decided to print the Y residuals for both the mean and median cases and to omit the percent deviations.

Table 5 summarizes the statistical calculations used in the two models.

FUTURE STUDIES

At the present time, attention is being given to one problem inherent in the exponential model (additive case): how to search a region of data points for the solution that gives the absolute minimum for the sum of squares of Y deviations when a relative solution has been reached. This problem would exist if any of the values under the heading "2YCALC - Y" (Fig. 11) were to be negative.

An example of such a problem is depicted in Fig. 12 for a one-independent-variable case. Because of the grouping of the data, a relative minimum could be found for either a straight line ($\mu_1 = 1$),

Table 5

SUMMARY OF STATISTICAL CALCULATIONS USED IN COMPUTER PROGRAM

Measurement	Logarithmic Model[a] (Multiplicative Error Assumed)	Exponential Model[b] (Additive Error Assumed)
SEY: Standard error of estimate	Median Case: $SEY_i=\sqrt{Y^2_{i_{calc}}\cdot e^{\hat{\sigma}^2}\cdot\left(e^{\hat{\sigma}^2}-1\right)}$, where $\hat{\sigma}^2=\frac{\sum_{i=1}^{N}(\ln Y_{i_{input}}-\ln Y_{i_{calc}})^2}{N-(p+1)}$. Mean Case: $SEY_i=\sqrt{Y^2_{i_{calc}}\cdot\left(e^{\hat{\sigma}^2}-1\right)}$, where $\hat{\sigma}^2=\frac{\sum_{i=1}^{N}(\ln Y_{i_{input}}-\ln Y_{i_{calc}})^2}{N-(p+1)}$.	$SEY=\sqrt{\frac{\sum_{i=1}^{N}(Y_{i_{input}}-Y_{i_{calc}})^2}{N-(p+1)}}$, where N = number of data points, p = number of parameters.
CV: Coefficient of variation (percent)	$CV_i=\left(\frac{SEY_i}{Y_{mean}}\right)\cdot 100$, where $Y_{mean}=\frac{\sum_{i=1}^{N}Y_{i_{input}}}{N}$.	$CV=\left(\frac{SEY}{Y_{mean}}\right)\cdot 100$, where $Y_{mean}=\frac{\sum_{i=1}^{N}Y_{i_{input}}}{N}$.
R: Coefficient of correlation (unadjusted for sample size)	Not applicable.	$R=\sqrt{1-\left[\frac{\sum_{i=1}^{N}(Y_{i_{input}}-Y_{i_{calc}})^2}{\sum_{i=1}^{N}(Y_{i_{input}}-Y_{mean_{input}})^2}\right]}$.
SDEVY: Standard deviation of input Y	$SDEVY=\sqrt{\frac{\sum_{i=1}^{N}(Y_{i_{input}}-Y_{mean_{input}})^2}{N-1}}$.	
Significance test of parameters (A,B,C,D)	If $\left\lvert\frac{\text{Value of parameter}}{\text{Standard deviation of parameter}}\right\rvert \geq$ Student's t-value, then the value of the parameter is considered to be significant.	

[a]For the logarithmic model, SEY and CV vary with each data point i.

[b]For the exponential model, SEY and CV are constant over the range of data points.

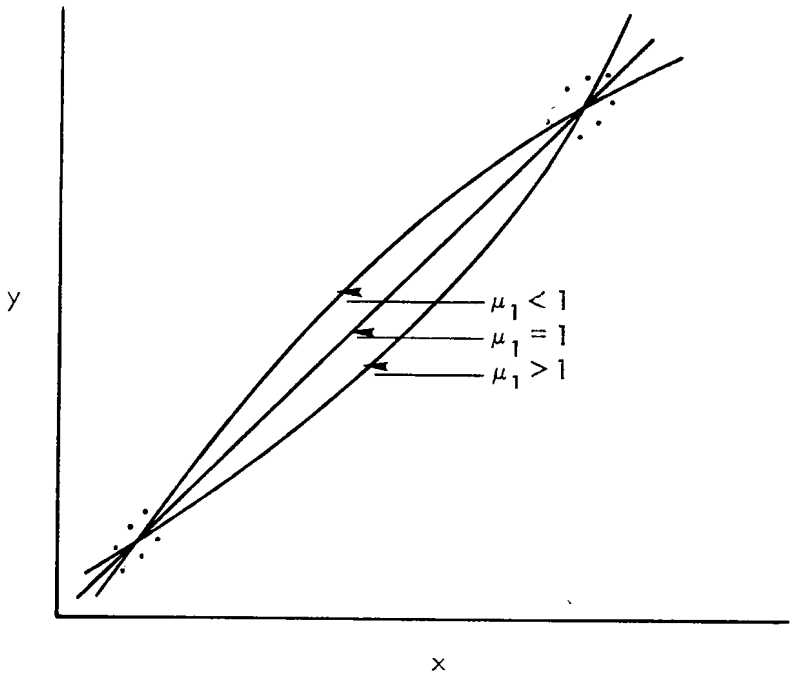


Fig. 12—Example of three fits through a set of data using curve of form $Y = e^{\mu_0} \cdot x^{\mu_1}$

a curved line bowed upward ($\mu_1 < 1$), or a curved line bowed downward ($\mu_1 > 1$), only one of which would be the absolute solution to the problem, i.e., the one with the minimum sum of squares of Y deviations. As more independent variables are considered, the problem becomes more complex, particularly for negative slopes.

The above problem illustrates the type of studies currently under consideration. Hopefully, as such studies are completed, their results can be incorporated into the computer program and documented. In this way, it is believed that this computer program may continue to reflect the latest methodologies relating to the general problem discussed in this Memorandum.

Appendix A

MATRIX DEFINITIONS AND OPERATIONS

This appendix gives definitions and operations of matrices that have been assumed in the body of this Memorandum.

MATRIX

An array of elements (usually real numbers) A with m rows and n columns is called a matrix of order (size) m × n. The elements are denoted by a_{ij}, where i refers to the number of the row and j refers to the number of the column in which the element appears. Thus

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & & a_{mn} \end{bmatrix} .$$

Any element a_{ij} with i = j is called a diagonal element of the matrix.

EQUALITY OF MATRICES

Equality of matrices is only defined for matrices of the same size. So consider two m × n matrices A and B with elements a_{ij} and b_{ij}, respectively. Then A = B if and only if

$$a_{ij} = b_{ij} \quad \text{for} \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n.$$

That is, each element of A must be equal to the corresponding element of B.

MATRIX OPERATIONS

Addition

This operation is also only defined for matrices of the same size, so let A and B be as above. Then C will be said to be the sum of A and B, written $C = A + B$, if C is an $m \times n$ matrix of elements c_{ij} obtained by

$$c_{ij} = a_{ij} + b_{ij} \quad \text{for} \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n.$$

Hence

$$C = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1n} + b_{1n} \\ a_{21} + b_{21} & a_{22} + b_{22} & \cdots & a_{2n} + b_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} + b_{m1} & a_{m2} + b_{m2} & \cdots & a_{mn} + b_{mn} \end{bmatrix}.$$

Scalar Multiplication

Let A be an $m \times n$ matrix of elements a_{ij} and let α be any fixed real number. Then the scalar multiplication of α and A, written αA, is given by multiplying each element of A by α. Hence

$$\alpha A = \alpha \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = \begin{bmatrix} \alpha a_{11} & \alpha a_{12} & \cdots & \alpha a_{1n} \\ \alpha a_{21} & \alpha a_{22} & \cdots & \alpha a_{2n} \\ \vdots & \vdots & & \vdots \\ \alpha a_{m1} & \alpha a_{m2} & \cdots & \alpha a_{mn} \end{bmatrix}.$$

Matrix Multiplication

Let A be an $m \times n$ matrix with elements a_{ij}, B be an $n \times p$ matrix with elements b_{ij}, and C be an $m \times p$ matrix with elements c_{ij}. Then

C is said to be equal to A times B, written C = AB, if

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \quad \text{for} \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, p.$$

Hence to obtain c_{ij}, the i-th row of A and the j-th column of B are multiplied together term by term and then added. Notice that the term-by-term multiplication cannot be accomplished without elements being left over unless the number of columns of A is equal to the number of rows of B. If this is not the case, then the multiplication is not defined. Notice also that the resulting matrix, C, has the same number of rows as A and the same number of columns as B, a consequence of the definition of multiplication. Finally it should be noted that $AB \neq BA$ in general. In fact, if $p \neq m$, then BA is not even defined. But even if $p = m$, it is still likely that $AB \neq BA$.

SPECIAL MATRICES

Square Matrix

A square matrix has the same number of rows as columns.

Identity Matrix.

An identity matrix is a square matrix whose diagonal elements are 1 and off-diagonal elements are 0. It is usually denoted by I. So the identity matrix of size 3 is given by

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Let I be the identity matrix of order n. Then it is the unique matrix that has the following properties:

$$AI = A \qquad \text{for all } m \times n \text{ matrices } A,$$

$$IB = B \qquad \text{for all } n \times m \text{ matrices } B,$$

$$CI = IC = C \qquad \text{for all } n \times n \text{ matrices } C.$$

Hence I is the matrix that corresponds to the number 1 in the real numbers, i.e., I has the properties in matrix multiplication that 1 has in real number multiplication.

Transpose Matrix

Let A be an $m \times n$ matrix with elements a_{ij}. Then the transpose of A, denoted by A', is defined to be the $n \times m$ matrix whose elements a'_{ij} are given by

$$a'_{ij} = a_{ji} \quad \text{for} \quad i = 1, \ldots, n; \quad j = 1, \ldots, m.$$

Hence the j-th row of A becomes the j-th column of A'.

Symmetric Matrix

A symmetric matrix is a square matrix that has the property that it is equal to its transpose. Hence if A is a square matrix of order n, then A is symmetric if and only if $A = A'$, i.e., if and only if $a_{ij} = a_{ji}$ for $i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, n$.

Inverse Matrix

An inverse matrix is defined only for square matrices, so let A be a square matrix of order n. Then the inverse of A, denoted by A^{-1},

is the unique square matrix of size n that has the properties that

$$A^{-1}A = AA^{-1} = I,$$

where I is the identity matrix of order n. Hence a square matrix is to its inverse as a real number is to its reciprocal. However, just as the reciprocal of zero is not defined (as a real number), so also A^{-1} is not always defined for every square matrix. If any row (column) of A is a linear combination of the remaining rows (columns), then the matrix is said to be singular, and its determinant is zero. In this case the inverse is not defined.

The method for obtaining the inverse will not be discussed here, as computer routines are usually available for this purpose. The explicit methods can be found in any book on matrix algebra, such as [4], for those who are interested.

Positive Definite Matrix

Let A be a square matrix of order n with real numbers for elements. Let $\tilde{a}$ be a row vector with n elements (real numbers), i.e.,

$$\tilde{a} = (a_1, \ldots, a_n).$$

Hence $\tilde{a}$ is a 1 × n matrix and $\tilde{a}'$ is then defined as the n × 1 matrix (column vector) given by

$$\tilde{a}' = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}.$$

Let the real valued function $f(\tilde{a})$ be defined by

$$f(\tilde{a}) = \tilde{a}\, A\, \tilde{a}'.$$

Now if A is such that $f(\tilde{a}) > 0$ for all nondegenerate $\tilde{a}$, i.e., $\tilde{a} \neq (0, 0, \ldots, 0)$, then A is said to be positive definite.

This concludes the matrix definitions and operations needed for this Memorandum.

Appendix B

STATISTICAL FACTS AND RELATIONSHIPS

This appendix is not meant to be a course in statistics but merely a statement of some of the statistical facts and relationships that have been used in this study and to which the reader might wish to refer. For clarification of terms used but not defined in this appendix or for further discussion of the ideas presented here, it is suggested that the reader refer to any good elementary book on mathematical statistics, such as Lindgren [5].

MEASURES OF CENTRAL TENDENCY

Two measures of central tendency have been used in this study. They are the mean and the median. Since the median is difficult to define uniquely for discrete distributions, a continuous random variable X with density f_X and cumulative distribution function F_X will be used to discuss the relationship between these two measures.

The Expected Value or Mean of X

The mean of X, denoted by E X, is defined by

$$E\ X = \int_{-\infty}^{\infty} X f_X(X)\ dX.$$

Hence it represents a weighted average of the values of the random variable by the probability of those values (i.e., $f_X(X)\ dX$).

Median of X

The median is obtained by finding the value, $X_{0.5}$, of the random

variable, X, that satisfies

$$\int_{-\infty}^{X_{0.5}} f_X(X)\ dX = \int_{X_{0.5}}^{\infty} f_X(X)\ dX = 0.5.$$

Hence it represents the middle of the distribution of X; i.e., the probability that $X < X_{0.5}$ is equal to 1/2 and the probability that $X > X_{0.5}$ is equal to 1/2.

That the two are not equal for all distributions should be clear from the definitions. The median depends on the value of the random variable only through the way in which the probability mass is spread over these values, while the mean depends on these values not only through the spread of the probability mass but also through the weighted average process. It is true, however, that these measures are equal for symmetric distributions such as the normal.

PERCENTILES OF A CONTINUOUS RANDOM VARIABLE

It should be noted that the median is just a special case of the more general concept of percentiles of a distribution. Using the notation of the previous section the percentiles of the distribution of X are defined as follows:

Choose any γ such that $0 < \gamma < 1$. Then the $\gamma(100)$-th percentile of X is that value X_γ that satisfies

$$\int_{-\infty}^{X_\gamma} f_X(X)\ dX = \gamma.$$

The significance of the $\gamma(100)$-th percentile is that the probability that $X < X_\gamma$ is equal to γ and the probability that $X > X_\gamma$ is equal to $1 - \gamma$. It is easy to see that the median is just the 50-th percentile of the distribution.

MEASURES OF VARIABILITY

Variance

The variance of a random variable X is denoted by Var X and is given by

$$\text{Var } X = E\,(X - E\,X)^2.$$

It is a measure of the dispersion of X about its expected value. A useful relationship is given by

$$\text{Var } X = E\,X^2 - [E\,X]^2.$$

Covariance

The covariance of two random variables X and Y is a measure of how they vary together. It is denoted by Cov (X, Y) and given by

$$\text{Cov } (X, Y) = E\,[(X - E\,X)(Y - E\,Y)].$$

If X - E X is large when Y - E Y is large, and small when Y - E Y is small, then X and Y will have a large positive covariance. If X - E X is small when Y - E Y is large, and large when Y - E Y is small, then the covariance of X and Y will be a large negative number. If one seems to have no effect on the other, then their covariance will be near zero. A useful relationship is given by

$$\text{Cov } (X, Y) = E\,XY - E\,X\,E\,Y.$$

From this it is easy to see that if X and Y are independent, then Cov (X, Y) = 0, since independence implies that E XY = E X E Y. This implication does not hold in the other direction, however.

Covariance Matrix

The multivariate generalization of the variance of a single random variable is given by the covariance matrix. If $X_1, X_2, \ldots, X_p$ have a multivariate distribution, then their covariance matrix is given by

$$\begin{bmatrix} \text{Var } X_1 & \text{Cov } (X_1, X_2) & \cdots & \text{Cov } (X_1, X_p) \\ \text{Cov } (X_2, X_1) & \text{Var } X_2 & \cdots & \text{Cov } (X_2, X_p) \\ \vdots & \vdots & & \vdots \\ \text{Cov } (X_p, X_1) & \text{Cov } (X_p, X_2) & \cdots & \text{Var } X_p \end{bmatrix}.$$

LINEAR COMBINATIONS OF RANDOM VARIABLES

The following equalities will help in computing the expected value, variance, and covariances of a linear combination of random variables. Let X, Y, and Z be random variables and a and b be real numbers. Then

$$E\,(aX + b) = a\,E\,X + b \quad \text{and} \quad E\,(X + Y) = E\,X + E\,Y;$$

$$\text{Var}\,(aX + b) = a^2\,\text{Var}\,X \quad \text{and} \quad \text{Var}\,(X + Y) = \text{Var}\,X + \text{Var}\,Y + 2\,\text{Cov}\,(X, Y);$$

$$\text{Cov}\,(aX + b, Z) = a\,\text{Cov}\,(X, Z) \quad \text{and} \quad \text{Cov}\,(X + Y, Z) = \text{Cov}\,(X, Z) + \text{Cov}\,(Y, Z).$$

Furthermore, if the random variables in question are all normal, then any linear combination of them is also normal.

DISTRIBUTIONS RELATED TO THE NORMAL

Chi-square Distribution

Let $X_1, X_2, \ldots, X_k$ be independent standard normal random variables (i.e., $E\,X_i = 0$ and $\text{Var}\,X_i = 1$ for $i = 1, 2, \ldots, k$). Let Z be the random variable defined by

$$Z = \sum_{i=1}^{k} X_i^2.$$

Then Z is a chi-square random variable with k degrees of freedom and

$$E\ Z = k \quad \text{and} \quad \text{Var}\ Z = 2k.$$

The t-distribution

Let X be a standard normal random variable and Z be a chi-square random variable with k degrees of freedom. Assume that X and Z are independent. Then the random variable T defined by

$$T = \frac{X}{\sqrt{Z/K}}$$

has a t-distribution with k degrees of freedom. It should be noted that the t-distribution is symmetric about the origin. Hence for any γ, $0 < \gamma < 1$, it follows that

$$t_{\gamma} = -t_{1-\gamma},$$

where t_{γ} and $t_{1-\gamma}$ are the $\gamma(100)$-th percentile and the $(1 - \gamma)(100)$-th percentile of the t-distribution.

Appendix C

CONVERSION OF e^{μ_0} TO $\bar{\mu}_0$

As mentioned in the Introduction, it is often desirable to have a regression function of the form

$$\bar{\mu}_0 X_1^{\mu_1} \cdots X_p^{\mu_p} \qquad (\bar{\mu}_0 > 0), \tag{1}$$

instead of a regression function of the form studied in this Memorandum, namely,

$$e^{\mu_0} X_1^{\mu_1} \cdots X_p^{\mu_p}. \tag{2}$$

This involves nothing more than making the transformation

$$\bar{\mu}_0 = e^{\mu_0}$$

in the original model. It seems only natural then to define the estimate of $\bar{\mu}_0$ by

$$\hat{\bar{\mu}}_0 = e^{\hat{\mu}_0}.$$

Furthermore, if this definition is adopted, then the values of the estimators $\hat{\mu}_1, \ldots, \hat{\mu}_p$, the distribution of these estimators, and the prediction interval obtained in the body of this Memorandum will remain valid for the new form of the regression function. So in adopting this definition, the only problem that remains in the conversion is to determine the distribution of $\hat{\bar{\mu}}_0$.

This can be accomplished by looking at the first order Taylor series expansion given by

$$e^{\hat{\mu}_0} \approx e^{\mu_0}[1 + (\hat{\mu}_0 - \mu_0)].$$

Then, by definition of $\bar{\mu}_0$, it follows that

$$\hat{\bar{\mu}}_0 \approx \bar{\mu}_0(1 + \hat{\mu}_0 - \mu_0),$$

which is a linear function of the random variable $\hat{\mu}_0$ since μ_0 and $\bar{\mu}_0$ are constants. But $\hat{\mu}_0$ was found to have a normal distribution (approximate for the additive case), so $\hat{\bar{\mu}}_0$ is approximately normal since it is approximately a linear function of $\hat{\mu}_0$. Furthermore,

$$E\ \hat{\bar{\mu}}_0 \approx \bar{\mu}_0(1 + E\ \hat{\mu}_0 - \mu_0)$$

and

$$E\ \hat{\mu}_0 = \mu_0$$

(approximately in the additive case). Hence

$$E\ \hat{\bar{\mu}}_0 \approx \bar{\mu}_0.$$

Similarly,

$$\text{Var}\ \hat{\bar{\mu}}_0 \approx (\bar{\mu}_0)^2\ \text{Var}\ \hat{\mu}_0 \approx \hat{\bar{\mu}}_0^2\ \text{Var}\ \hat{\mu}_0$$

and

$$\text{Cov}\ (\hat{\bar{\mu}}_0, \hat{\mu}_i) \approx \bar{\mu}_0\ \text{Cov}\ (\hat{\mu}_0, \hat{\mu}_i) \approx \hat{\bar{\mu}}_0\ \text{Cov}\ (\hat{\mu}_0, \hat{\mu}_i)$$

for $i = 1, 2, \ldots, p$.

Using these approximations for the distribution of $\hat{\bar{\mu}}_0$, the conversion is complete.

Appendix D

ABSOLUTE MINIMUM AND POSITIVE DEFINITENESS OF THE MATRICES OF SECOND PARTIALS

GENERAL DISCUSSION

Suppose we have a function, f, of one variable, say Z, and suppose that we are interested in finding the value of Z, say Z_0, that minimizes f(Z). With the usual continuity assumptions on f, the first thing that would be done is to find the roots of the equation

$$f'(Z) = 0,$$

where f' is the first derivative of f with respect to Z. Each root of this equation is called a critical point. As mentioned in the additive model, a critical point may represent a relative or absolute maximum or minimum or an inflection point. To find a critical point that represents a minimum, one finds the second derivative of f with respect to Z, denoted by $f''(Z)$. If this is positive when evaluated at the critical point in question, then that particular critical point is a relative minimum.

Take a relative minimum critical point and denote it by Z_0. Then

$$f'(Z_0) = 0 \quad \text{and} \quad f''(Z_0) > 0.$$

Now if $f''(Z) > 0$ for all Z, then Z_0 is the only critical point for f, and Z_0 is therefore the absolute minimum of f. Similarly, for any interval around Z_0, say, $Z_1 < Z_0 < Z_2$, Z_0 will be the only critical point of f in this interval and will therefore be the absolute minimum of f in this interval if $f''(Z) > 0$ for all Z in the interval, i.e., for all Z satisfying $Z_1 < Z < Z_2$.

The matrix of second partial derivatives is a generalization of the second derivative. Any critical point that has a positive definite matrix[†] of second partial derivatives, when evaluated at that critical point, is a relative minimum for the surface. If the matrix of second partials is positive definite for all points in a region including this critical point, then this critical point is an absolute minimum in that region and is the only critical point in that region.

SPECIFIC CASE

The matrix of second partials for Q is given below:

$$2\begin{bmatrix} \sum_{i=1}^{n} f_i(2f_i - Y_i) & \sum_{i=1}^{n} f_i \ln X_{1i}(2f_i - Y_i) & \sum_{i=1}^{n} f_i \ln X_{2i}(2f_i - Y_i) & \cdots & \sum_{i=1}^{n} f_i \ln X_{pi}(2f_i - Y_i) \\ \sum_{i=1}^{n} f_i \ln X_{1i}(2f_i - Y_i) & \sum_{i=1}^{n} f_i \ln^2 X_{1i}(2f_i - Y_i) & \sum_{i=1}^{n} f_i \ln X_{1i} \ln X_{2i}(2f_i - Y_i) & \cdots & \sum_{i=1}^{n} f_i \ln X_{1i} \ln X_{pi}(2f_i - Y_i) \\ \vdots & \vdots & \vdots & & \vdots \\ \sum_{i=1}^{n} f_i \ln X_{pi}(2f_i - Y_i) & \sum_{i=1}^{n} f_i \ln X_{pi} \ln X_{1i}(2f_i - Y_i) & \sum_{i=1}^{n} f_i \ln X_{pi} \ln X_{2i}(2f_i - Y_i) & \cdots & \sum_{i=1}^{n} f_i \ln^2 X_{pi}(2f_i - Y_i) \end{bmatrix}.$$

In this matrix, $f_i = e^{\mu_0} X_{1i}^{\mu_1} \cdots X_{pi}^{\mu_p}$ for i = 1, 2, ..., n, and the element in the k-th row and the j-th column is

$$\frac{\partial Q^2}{\partial \mu_k \partial \mu_j}.$$

It is assumed that this matrix is nonsingular. Now assume that

[†]For definition, see Appendix A.

$2f_i - Y_i > 0$ for all i; then it is possible to make the following substitutions. For i = 1, 2, ..., n, and j = 1, 2, ..., p, let

$$\eta_{0i}^2 = f_i(2f_i - Y_i),$$

$$\eta_{ji}^2 = f_i \ln^2 X_{ji}(2f_i - Y_i).$$

Then the matrix of second partials becomes

$$C = \begin{bmatrix} \sum_{i=1}^{n} \eta_{0i}^2 & \sum_{i=1}^{n} \eta_{0i}\eta_{1i} & \sum_{i=1}^{n} \eta_{0i}\eta_{2i} & \cdots & \sum_{i=1}^{n} \eta_{0i}\eta_{pi} \\ \sum_{i=1}^{n} \eta_{1i}\eta_{0i} & \sum_{i=1}^{n} \eta_{1i}^2 & \sum_{i=1}^{n} \eta_{1i}\eta_{2i} & \cdots & \sum_{i=1}^{n} \eta_{1i}\eta_{pi} \\ \vdots & \vdots & \vdots & & \vdots \\ \sum_{i=1}^{n} \eta_{pi}\eta_{0i} & \sum_{i=1}^{n} \eta_{pi}\eta_{1i} & \sum_{i=1}^{n} \eta_{pi}\eta_{2i} & \cdots & \sum_{i=1}^{n} \eta_{pi}^2 \end{bmatrix}.$$

Define vectors $\eta_i = (\eta_{0i}, \eta_{1i}, \ldots, \eta_{pi})$ for i = 1, 2, ..., n. Then C can be expressed as

$$C = \sum_{i=1}^{n} \eta_i' \eta_i,$$

where

$$\eta_i' = \begin{bmatrix} \eta_{0i} \\ \eta_{1i} \\ \vdots \\ \eta_{pi} \end{bmatrix},$$

the transpose of η_i. Now take any p + 1 dimensional row vector $\tilde{a} = (a_0, a_1, \ldots, a_p)$. Then

$$\tilde{a}C\tilde{a}' = \sum_{i=1}^{n} (\tilde{a}\eta_i')(\eta_i\tilde{a}')$$

$$= \sum_{i=1}^{n} b_1^2,$$

where

$$b_i = \sum_{j=1}^{n} a_j \eta_{ji} = \tilde{a}\eta_i' = \eta_i\tilde{a}'.$$

Therefore, $\tilde{a}C\tilde{a}' \geq 0$ for all $\tilde{a}$, since it is really the sum of squares. Furthermore, equality holds only if $b_i = 0$ for $i = 1, 2, \ldots, n$. This can happen only if

(i) $\tilde{a}$ is degenerate, i.e., $\tilde{a} = (0, \ldots, 0)$, or

(ii) the rows of C are linearly dependent, i.e., there exists an $\tilde{a}$ such that $\tilde{a}\eta_i' = 0$ for $i = 1, 2, \ldots, p$.

But (ii) cannot happen, since this would imply that the matrix C is singular, which in turn would imply that the matrix of second partial derivatives is singular.

Therefore for any nondegenerate $\tilde{a}$ we have that

$$\tilde{a}C\tilde{a}' > 0,$$

and hence C is positive definite. So if $2f_i - Y_i > 0$ for all i, then the matrix of second partial derivatives of Q is positive definite.

Now take a given set of historical data. Then $2f_i - Y_i > 0$ for all i on a certain region R of the parameters $\mu_0, \ldots, \mu_p$. This region then depends on the historical data. If $(\hat{\mu}_0, \ldots, \hat{\mu}_p)$ is in this region, then it corresponds to the critical point Z_0 in the one-variable model. The region R corresponds to the interval given by (Z_1, Z_2). Therefore $(\hat{\mu}_0, \ldots, \hat{\mu}_p)$ represents the only critical point in this region, and it is the absolute minimum for Q in this region.

Appendix E

VARIANCE-COVARIANCE MATRIX

The variance-covariance matrix of the estimates $(\hat{\mu}_0, \ldots, \hat{\mu}_p)$ is obtained by a direct application of the Gabler-Madansky study [2]. To translate the notation of that paper into the notation of this work the following comparative listing should be of value.

Graver-Boren	Gabler-Madansky
n	p
$p + 1$	t
$(Y_1, \ldots, Y_n)$	$(X_1, \ldots, X_p)$
$(\mu_0, \ldots, \mu_p)$	$(\mu_1, \ldots, \mu_t)$
σ_A^2	Ω

The first thing that must be done is to establish the $p + 1$ linear relations referred to in [2] by

$$f_k(\alpha, \mu) = 0 \quad \text{for } k = 0, 1, \ldots, p.$$

As one can see, each f is a function of α and μ. The latter refers to the parameters $\mu_0, \mu_1, \ldots, \mu_p$, while the former is defined by

$$\alpha_k = e^{\mu_0} X_{1k}^{\mu_1} \cdots X_{pk}^{\mu_p} \quad \text{for } k = 1, 2, \ldots, n.$$

Inherent in the model discussed in this Memorandum is the relationship

$$\underline{\alpha} = B\underline{\mu} \ ,$$

where

$$\underline{\alpha} = \begin{bmatrix} \ln \alpha_1 \\ \vdots \\ \ln \alpha_n \end{bmatrix},$$

$$B = \begin{bmatrix} 1 & \ln X_{11} & \ln X_{21} & \cdots & \ln X_{p1} \\ 1 & \ln X_{12} & \ln X_{22} & \cdots & \ln X_{p2} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \ln X_{1n} & \ln X_{2n} & \cdots & \ln X_{pn} \end{bmatrix},$$

$$\underline{\mu} = \begin{bmatrix} \mu_0 \\ \mu_1 \\ \vdots \\ \mu_p \end{bmatrix}.$$

This implies

$$B'\underline{\alpha} = B'B\underline{\mu}$$

or

$$(B'B)^{-1}B'\underline{\alpha} = \underline{\mu} ,$$

provided $B'B$ is nonsingular. Hence, we have

$$\underline{\mu} - (B'B)^{-1}B'\underline{\alpha} = \underline{0} .$$

The last set of equations describes the p + 1 relationships referred to above.

Gabler and Madansky also make use of n - (p + 1) relationships referred to by

$$\psi_k(\alpha) = 0 \qquad \text{for } k = 1, 2, \ldots, n - (p + 1) .$$

Note that these relationships depend explicitly on only the α's. Care must be taken to use each of the n α's equally in setting up these relationships to avoid obtaining estimates of the variance-covariance matrix that depend on the order of the sample. Equal consideration of the α's can be accomplished by making use of eigenvalues in the reduction of a size n matrix to a size n - (p + 1) matrix. The use of this technique can be seen in the following derivation of $\psi_k(\alpha) = 0$, $k = 1, \ldots, n - (p + 1)$.

Two relationships have already been established. They are

$$\underline{\alpha} = B\underline{\mu}$$

and

$$\underline{\mu} = (B'B)^{-1}B'\underline{\alpha} .$$

By substitution we have

$$\underline{\alpha} = B(B'B)^{-1}B'\underline{\alpha}$$

or

$$\underline{0} = (I - B(B'B)^{-1}B')\underline{\alpha} ,$$

where I is the n × n identity matrix.

Let $M = (I - B(B'B)^{-1}B')$ and $E = B(B'B)^{-1}B'$. Then $M\underline{\alpha} = \underline{0}$ and $M = I - E$. However, M and E are idempotent,[†] which implies that the eigenvalues are either 0 or 1. Since B is an n × (p + 1) matrix, its rank is at most p + 1 and is equal to p + 1 if B'B is nonsingular. In this case the rank of E is p + 1. This implies that E has p + 1 eigenvalues equal to 1, and n - (p + 1) eigenvalues equal to 0. This in turn implies that M has n - (p + 1) eigenvalues equal to 1, and p + 1 eigenvalues equal to 0. Thus $M = QI^{*}Q'$, where Q is the n × n orthogonal[††] matrix of eigenvectors, and

$$I^{*} = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix},$$

where I is the n - (p + 1) × n - (p + 1) identity matrix. However, $\underline{0} = M\underline{\alpha} = QI^{*}Q'\underline{\alpha}$ implies that $I^{*}Q'\underline{\alpha} = \underline{0}$. This result is analogous to the one-dimensional statement that if (a)(b) = 0 and $a \neq 0$, then b = 0 since Q orthogonal is analogous to $a \neq 0$. Therefore, $I^{*}Q'\underline{\alpha} = \underline{0}$ can be rewritten vectorially as

$$Q^{*}\underline{\alpha} = (Q_1', Q_3')\underline{\alpha} = \underline{0} ,$$

where

[†]Note that a matrix A is idempotent if and only if $A^2 = A$.

[††]Note that Q being orthogonal implies that $QQ' = I$, the n × n identity matrix. Hence Q is of full rank.

$$Q' = \left[\begin{array}{c|c} \overset{n-(p+1)}{Q_1'} & \overset{p+1}{Q_3'} \\ \hline Q_2' & Q_4' \end{array}\right] \begin{matrix} \\ \\ \end{matrix} \begin{matrix} n-(p+1) \\ \\ p+1 \end{matrix}$$

Now let $\psi_k(\alpha) = \sum_{j=1}^{n} q^*_{kj} \ln \alpha_j = 0$ for $k = 1, 2, \ldots, n - (p + 1)$, where q^*_{kj} is the element in the k-th row and j-th column of $Q^* = (Q_1', Q_3')$.

Referring again to the Gabler-Madansky model, we find that the variance-covariance matrix is given by

$$\sigma^2 R = \sigma^2 C'AC \ .$$

In the above, $A = I - \psi_\alpha'(\psi_\alpha \psi_\alpha')^{-1}\psi_\alpha$ with ψ_α an $n - (p + 1) \times n$ matrix of elements

$$\psi_{\alpha kj} = \frac{\partial \psi_k(\alpha)}{\partial \alpha_j} \ ;$$

C is the $n \times (p + 1)$ matrix given by

$$C = -(F_\mu^{-1} F_\alpha)' \ ,$$

where F_α is a $p + 1 \times n$ matrix with elements given by

$$f_{\alpha kj} = \frac{\partial f_k(\alpha, \mu)}{\partial \alpha_j} \ ,$$

and F_μ is the $p + 1 \times p + 1$ matrix whose elements are given by

$$f_{\mu kj} = \frac{\partial f_k(\alpha, \mu)}{\partial \mu_j} \ .$$

In our case,

$$\psi_{\alpha kj} = \frac{q^{*}_{kj}}{\alpha_j} ,$$

$$F_{\mu} = I ,$$

$$f_{\alpha jk} = \frac{h_{jk}}{\alpha_k} ,$$

where h_{jk} is an element of

$$H = (B'B)^{-1}B' .$$

Therefore, C' = a matrix whose elements are given by $c_{kj} = h_{kj}/\alpha_j$.

Since A and C are now defined, we have only to calculate

$$R = C'AC.$$

However, the eigenvalues of A have been used in the computer to enhance the accuracy of the calculation of A.

Noting that the matrix $P = \psi'_{\alpha}(\psi_{\alpha}\psi'_{\alpha})^{-1}\psi_{\alpha}$ is idempotent (i.e., $P^2 = P$), we see that the eigenvalues of P equal either 0 or 1. Since ψ_{α} is an $n - (p + 1) \times n$ matrix, its rank is at most $n - (p + 1)$, and is equal to $n - (p + 1)$ if $\psi_{\alpha}\psi'_{\alpha}$ is nonsingular. In this case, the rank of P is $n - (p + 1)$, so that P has exactly $n - (p + 1)$ eigenvalues equal to 1, and its other $p + 1$ eigenvalues are 0. Consequently, $A = I - P$ has $p + 1$ eigenvalues equal to 1 and $n - (p + 1)$ eigenvalues equal to 0.

Since $p + 1$ is much smaller than n, replacing the matrix A by QQ', where Q is the $n \times p + 1$ matrix of eigenvectors corresponding to the $p + 1$ eigenvalues equal to 1 (or better yet, by QDQ', where D is the $p + 1 \times p + 1$ diagonal matrix of the largest $p + 1$ eigenvalues of A, as computed complete with numerical rounding error, so that D does not quite equal I) will increase the accuracy of the computation of $C'AC$.

Appendix F

FORTRAN IV SYMBOLIC PROGRAM

This appendix presents a symbolic listing of the FORTRAN IV computer program. However, the two auxiliary subroutines (SETSW and FPTCTL) are written in MAP language specifically for the RAND IBM 7040/7044 computer, and if they are to be used for other computer systems, they may have to be rewritten. It should be noted that these two subroutines are not essential to the operation of the computer program. They are used to provide a way to retain control of the machine if overflows occur.

```
$IBFTC MAIN
      COMMON  A,          B,          C,          D,          A1,
     C        CV,         PDEVM,      R,          RN,         RN1,
     C        SDEVA,      SDEVA1,     SDEVB,      SDEVC,      SDEVD,
     C        SDEVY,      SEY,        SIGYSQ,     SUMDS1,     SUMDS2,
     C        SUMY,       SUMYSQ,     T,          XLS,        YMEAN
      COMMON  YMS,        I1,         IERR,       IEXPO,      IDESIG,
     C        K1,         KM,         LS,         M,          MN,
     C        N,          COV,        G,          H,          ISC,
     C        RSC,        Q,          PDEV,       UP,         XLP,
     C        WL,         XL,         ZL,         YL,         YC
      COMMON  YC1,        YDEV,       YDEV1,      W,          X,
     C        Z,          EM,         YSD,        YSD1,       TITLE
      COMMON  AN,         EA,         EA1,        IEIG,       IMAT
      DIMENSION  COV(4,4), G(50,50), H(50,50), ISC(4), RSC(4), Q(50,1),
     C IDENT(50), PDEV(50), UP(50), XLP(50), WL(50), XL(50), ZL(50),
     C YL(50), YC(50), YC1(50), YDEV(50), YDEV1(50), W(50), X(50),
     C Z(50), Y(50) , W1(50), X1(50), Z1(50), EM(50,5), YSD(50),
     C YSD1(50), TITLE(10), F(1)
      EQUIVALENCE  (F(1), A), (IDENT(1), EM(1,1)), (Y(1), EM(1,2))
      EQUIVALENCE  (W1(1), EM(1,3)), (X1(1), EM(1,4)), (Z1(1), EM(1,5))
C
C     MAIN ROUTINE OF MULTIVARIATE LOGARITHMIC AND EXPONENTIAL COMPUTER
C      REGRESSION PROGRAM
C
C     SET SWITCHES SO THAT EXCESSIVE FLOATING-POINT TRAPS ARE IGNORED.
    1 CALL SETSW (0, 1, 0, 0)
C     CLEAR ALL STORAGE LOCATIONS IN COMMON.
    2 DO 3 I = 1, 6175
    3 F(I) = 0.
C     SET PASS COUNTER TO 1.
      M = 1
    4 CALL READ
C     CHECK ERROR DESIGNATOR.
      IF (IERR .EQ. 1) GO TO 2
      CALL ARRAY
      CALL CALC
C     CHECK ERROR DESIGNATOR.
      IF (IERR .EQ. 1) GO TO 5
      CALL EXPO
C     CHECK ERROR DESIGNATOR.
      IF (IERR .EQ. 1) GO TO 5
      CALL STATIS
    5 IERR = 0
      IF (M .GE. 7) GO TO 2
C     INCREASE PASS COUNTER BY 1
      M = M + 1
      GO TO 4
      END
```

```
$IBFTC READ
      SUBROUTINE READ
      COMMON  A,          B,          C,          D,          A1,
     C        CV,         PDEVM,      R,          RN,         RN1,
     C        SDEVA,      SDEVA1,     SDEVB,      SDEVC,      SDEVD,
     C        SDEVY,      SEY,        SIGYSQ,     SUMDS1,     SUMDS2,
     C        SUMY,       SUMYSQ,     T,          XLS,        YMEAN
      COMMON  YMS,        I1,         IERR,       IEXPO,      IDESIG,
     C        K1,         KM,         LS,         M,          MN,
     C        N,          COV,        G,          H,          ISC,
     C        RSC,        Q,          PDEV,       UP,         XLP,
     C        WL,         XL,         ZL,         YL,         YC
      COMMON  YC1,        YDEV,       YDEV1,      W,          X,
     C        Z,          EM,         YSD,        YSD1,       TITLE
      COMMON  AN,         EA,         EA1,        IEIG,       IMAT
      DIMENSION  COV(4,4), G(50,50), H(50,50), ISC(4), RSC(4), Q(50,1),
     C IDENT(50), PDEV(50), UP(50), XLP(50), WL(50), XL(50), ZL(50),
     C YL(50), YC(50), YC1(50), YDEV(50), YDEV1(50), W(50), X(50),
     C Z(50), Y(50) , W1(50), X1(50), Z1(50), EM(50,5), YSD(50),
     C YSD1(50), TITLE(10), F(1)
      DIMENSION  FMT(12), JX(5)
      EQUIVALENCE  (F(1), A), (IDENT(1), EM(1,1)), (Y(1), EM(1,2))
      EQUIVALENCE  (W1(1), EM(1,3)), (X1(1), EM(1,4)), (Z1(1), EM(1,5))
      EQUIVALENCE  (JX(1), J1), (JX(2), J2), (JX(3), J3), (JX(4), J4)
      EQUIVALENCE  (JX(5), J5)
C
C     SUBROUTINE FOR READING INPUT DATA, CHECKING FOR ERRORS, AND
C      SCALING DATA IF REQUIRED
C
    1 IF (M .GT. 1) GO TO 26
C     READ TITLE CARD.
      READ (5, 2) IFMT, IDESIG, LS, (ISC(I), I = 1, 4), (TITLE(I) , I =
     C 1, 10)
    2 FORMAT (3I1, 1X, 4I2, 2X, 9A6, A4)
C     TEST WHETHER FORMAT CARD IS TO BE READ FOR THIS SET OF DATA.
      IF (IFMT .EQ. 0) GO TO 26
C     READ ORDER CARD.
      READ (5, 20) J1, J2, J3, J4, J5
   20 FORMAT (5(A2, 1X))
      DO 5 K = 1, 5
      CALL SEARCH (JX(K),L,36HID    Y1    X1    X2    X3          )
      IF (L .EQ. 7) GO TO 22
    5 JX(K) = L
      GO TO 24
C     WRITE ERROR MESSAGE
   22 WRITE (6, 23)
   23 FORMAT(1H1//, 5X,  82HERROR IN VARIABLE FORMAT INDEX (SECOND INPUT
     C CARD).  THIS JOB HAS BEEN TERMINATED.)
      CALL EXIT
C     READ FORMAT CARD.
   24 READ (5, 25) (FMT(I), I = 1, 12)
   25 FORMAT (12A6)
C     PRINT TITLE ON NEW PAGE.
   26 WRITE (6, 3) (TITLE(I), I = 1, 10)
    3 FORMAT (1H1, 9X, 9A6, A4 /)
      IF (M .GT. 1) GO TO 12
```

```
C       COMPUTE SCALE FACTORS (IF ENTERED) FOR VARIABLES.
        DO 4 I = 1, 4
        IF (ISC(I) .NE. 0) RSC(I) = 10.0**(ISC(I))
      4 CONTINUE
C       IF NOT ENTERED, SET LEVEL-OF-SIGNIFICANCE DESIGNATOR TO 2
C        (LOS OF 0.05).
        IF (LS .EQ. 0) LS = 2
C       READ INPUT DATA.
        DO 6 I = 1, 51
        IF (J5 .EQ. 6) GO TO 27
        READ (5, FMT) EM(I,J1), EM(I,J2), EM(I,J3), EM(I,J4), EM(I,J5)
        GO TO 30
     27 IF (J4 .EQ. 6) GO TO 28
        READ (5, FMT) EM(I,J1), EM(I,J2), EM(I,J3), EM(I,J4)
        GO TO 30
     28 IF (J3 .EQ. 6) GO TO 29
        READ (5, FMT) EM(I,J1), EM(I,J2), EM(I,J3)
        GO TO 30
     29 READ (5, FMT) EM(I,J1), EM(I,J2)
C       CHECK FOR BLANK CARD.
     30 IF (ABS(Y(I)) + ABS(W1(I)) + ABS(X1(I)) + ABS(Z1(I)) .EQ. 0.)
          C GO TO 7
      6 CONTINUE
        I = 51
C       SET N EQUAL TO NUMBER OF INPUT DATA POINTS.
      7 N = I - 1
        DO 9 I = 1, N
C       CHECK INPUT DATA FOR ERRORS.
        IF (Y(I) .LE. 0.) GO TO 10
        IF (W1(I) .LE. 0.) GO TO 10
        IF (X1(I) .LT. 0.) GO TO 10
        IF (Z1(I) .LT. 0.) GO TO 10
C       SCALE INPUT DATA IF SCALE FACTORS ARE ENTERED.
      8 IF (ISC(1) .NE. 0) Y(I) = Y(I) * RSC(1)
        IF (ISC(2) .NE. 0) W1(I) = W1(I) * RSC(2)
        IF (ISC(3) .NE. 0) X1(I) = X1(I) * RSC(3)
        IF (ISC(4) .NE. 0) Z1(I) = Z1(I) * RSC(4)
      9 CONTINUE
        GO TO 12
C       PRINT ERROR MESSAGE IF ANY ERRORS EXIST IN INPUT DATA.
     10 WRITE (6, 11)
     11 FORMAT (1HO/, 5X, 109HZERO OR NEGATIVE VALUE EXISTS IN INPUT DATA.
       C  ALL RUNS ASSOCIATED WITH THIS SET OF DATA HAVE BEEN TERMINATED.)
        IERR = 1
        RETURN
C       PRINT SCALE FACTORS IF ENTERED.
     12 IF (ISC(1) .NE. 0) WRITE (6, 13) ISC(1)
        IF (ISC(2) .NE. 0) WRITE (6, 14) ISC(2)
        IF (ISC(3) .NE. 0) WRITE (6, 15) ISC(3)
        IF (ISC(4) .NE. 0) WRITE (6, 16) ISC(4)
     13 FORMAT (1HO, 9X, 36HY SCALE FACTOR  =  BASE 10 EXPONENT , I2)
     14 FORMAT (1HO, 9X, 37HX1 SCALE FACTOR  =  BASE 10 EXPONENT , I2)
     15 FORMAT (1HO, 9X, 37HX2 SCALE FACTOR  =  BASE 10 EXPONENT , I2)
     16 FORMAT (1HO, 9X, 37HX3 SCALE FACTOR  =. BASE 10 EXPONENT , I2)
        RETURN
        END
```

```
$IBFTC ARRAY
       SUBROUTINE ARRAY
       COMMON  A,          B,          C,          D,          A1,
      C        CV,         PDEVM,      R,          RN,         RN1,
      C        SDEVA,      SDEVA1,     SDEVB,      SDEVC,      SDEVD,
      C        SDEVY,      SEY,        SIGYSQ,     SUMDS1,     SUMDS2,
      C        SUMY,       SUMYSQ,     T,          XLS,        YMEAN
       COMMON  YMS,        I1,         IERR,       IEXPO,      IDESIG,
      C        K1,         KM,         LS,         M,          MN,
      C        N,          COV,        G,          H,          ISC,
      C        RSC,        Q,          PDEV,       UP,         XLP,
      C        WL,         XL,         ZL,         YL,         YC
       COMMON  YC1,        YDEV,       YDEV1,      W,          X,
      C        Z,          EM,         YSD,        YSD1,       TITLE
       COMMON  AN,         EA,         EA1,        IEIG,       IMAT
       DIMENSION  COV(4,4), G(50,50), H(50,50), ISC(4), RSC(4), Q(50,1),
      C IDENT(50), PDEV(50), UP(50), XLP(50), WL(50), XL(50), ZL(50),
      C YL(50), YC(50), YC1(50), YDEV(50), YDEV1(50), W(50), X(50),
      C Z(50), Y(50) , W1(50), X1(50), Z1(50), EM(50,5), YSD(50),
      C YSD1(50), TITLE(10), F(1)
       EQUIVALENCE  (F(1), A), (IDENT(1), EM(1,1)), (Y(1), EM(1,2))
       EQUIVALENCE  (W1(1), EM(1,3)), (X1(1), EM(1,4)), (Z1(1), EM(1,5))
C
C      SUBROUTINE FOR ORDERING THE DATA AND SELECTING APPROPRIATE SET(S)
C       OF INDEPENDENT VARIABLES TO BE USED FOR THIS PASS THROUGH THE
C       PROGRAM
C
    1  IF (M .GT. 1) GO TO 30
C      ORDER DATA FROM LOWEST Y VALUE TO HIGHEST Y VALUE.
       NK = N - 1
       DO 29 I = 1, NK
       IN = I + 1
       DO 2 J = IN, N
       IF (Y(I) .LE. Y(J)) GO TO 2
       TEMP = Y(I)
       Y(I) = Y(J)
       Y(J) = TEMP
       TEMP = W1(I)
       W1(I) = W1(J)
       W1(J) = TEMP
       TEMP = X1(I)
       X1(I) = X1(J)
       X1(J) = TEMP
       TEMP = Z1(I)
       Z1(I) = Z1(J)
       Z1(J) = TEMP
       ITEMP = IDENT(I)
       IDENT(I) = IDENT(J)
       IDENT(J) = ITEMP
    2  CONTINUE
   29  CONTINUE
C      IF IDESIG IS EQUAL TO 1, SET M TO 7 SO THAT ONLY ONE PASS WILL
C       BE MADE THROUGH PROGRAM.
   30  IF (IDESIG .EQ. 1) M = 7
    4  IF (Z1(1) .GT. 0.) GO TO 5
       IF (X1(1) .GT. 0.) GO TO 20
```

```
      M = 7
      GO TO 6
    5 IF (IDESIG .EQ. 1) GO TO 18
      GO TO (6, 8, 10, 12, 14, 16, 18), M
C     SET VALUES OF PROGRAM INDEPENDENT VARIABLES (W, X, Z) TO VALUES OF
C      APPROPRIATE INPUT INDEPENDENT VARIABLES (W1, X1, Z1) DEPENDING
C      ON WHICH SETS OF INDEPENDENT VARIABLES ARE BEING CONSIDERED FOR
C      THIS PASS. ALSO SET I1 EQUAL TO NUMBER OF INDEPENDENT VARIABLES
C      BEING CONSIDERED FOR THIS PASS.
    6 DO 7 I = 1, N
    7 W(I) = W1(I)
      I1 = 1
      RETURN
    8 DO 9 I = 1, N
    9 W(I) = X1(I)
      I1 = 1
      RETURN
   10 DO 11 I = 1, N
   11 W(I) = Z1(I)
      I1 = 1
      RETURN
   12 DO 13 I = 1, N
      W(I) = W1(I)
   13 X(I) = X1(I)
      I1 = 2
      RETURN
   14 DO 15 I = 1, N
      W(I) = W1(I)
   15 X(I) = Z1(I)
      I1 = 2
      RETURN
   16 DO 17 I = 1, N
      W(I) = X1(I)
   17 X(I) = Z1(I)
      I1 = 2
      RETURN
   18 DO 19 I = 1, N
      W(I) = W1(I)
      X(I) = X1(I)
   19 Z(I) = Z1(I)
      I1 = 3
      RETURN
   20 IF (IDESIG .EQ. 1) GO TO 12
      M = M + 2
   21 GO TO (6, 6, 6, 6, 6, 8, 8, 8, 12), M
      END
```

```
$IBFTC CALC
      SUBROUTINE CALC
      COMMON   A,          B,          C,          D,          A1,
     C         CV,         PDEVM,      R,          RN,         RN1,
     C         SDEVA,      SDEVA1,     SDEVB,      SDEVC,      SDEVD,
     C         SDEVY,      SEY,        SIGYSQ,     SUMDS1,     SUMDS2,
     C         SUMY,       SUMYSQ,     T,          XLS,        YMEAN
      COMMON   YMS,        I1,         IERR,       IEXPO,      IDESIG,
     C         K1,         KM,         LS,         M,          MN,
     C         N,          COV,        G,          H,          ISC,
     C         RSC,        Q,          PDEV,       UP,         XLP,
     C         WL,         XL,         ZL,         YL,         YC
      COMMON   YC1,        YDEV,       YDEV1,      W,          X,
     C         Z,          EM,         YSD,        YSD1,       TITLE
      COMMON   AN,         EA,         EA1,        IEIG,       IMAT
      COMMON   /LABEL/ S, VAR
      DIMENSION  COV(4,4), G(50,50), H(50,50), ISC(4), RSC(4), Q(50,1),
     C IDENT(50), PDEV(50), UP(50), XLP(50), WL(50), XL(50), ZL(50),
     C YL(50), YC(50), YC1(50), YDEV(50), YDEV1(50), W(50), X(50),
     C Z(50), Y(50) , W1(50), X1(50), Z1(50), EM(50,5), YSD(50),
     C YSD1(50), TITLE(10), F(1)
      DIMENSION  TT(1), PIVOT(50), IPIVOT(50), INDEX(50,2)
      EQUIVALENCE  (F(1), A), (IDENT(1), EM(1,1)), (Y(1), EM(1,2))
      EQUIVALENCE  (W1(1), EM(1,3)), (X1(1), EM(1,4)), (Z1(1), EM(1,5))
C
C     SUBROUTINE FOR OBTAINING SOLUTIONS AND RELATED STATISTICS FOR
C      LOGARITHMIC CASE
C
C     USE MODIFIED FLOATING-POINT TRAP ROUTINE FTPCTL SO THAT CONTROL
C      IS RETAINED BY THE PROGRAM IF EXCESSIVE FLOATING-POINT TRAPS ARE
C      ENCOUNTERED IN THE CALCULATIONS.
    1 ASSIGN 56 TO III
      CALL FPTCTL(III)
C     CASE DESIGNATOR (LOGARITHMIC -- IEXPO = 0,
C      EXPONENTIAL -- IEXPO = 1)
      IEXPO = 0
C     NUMBER OF PARAMETERS
      MN = I1 + 1
C     DEGREES OF FREEDOM
      KM = N - MN
C     CHECK WHETHER DEGREES OF FREEDOM ARE GREATER THAN ZERO. IF NOT,
C      PRINT ERROR MESSAGE.
      IF (KM .LE. 0) GO TO 54
C     FLOAT N AND KM.
      AN = N
      RKM = KM
C     PRINT HEADING.
      WRITE (6, 3)
    3 FORMAT (1HO, 59X, 16HLOGARITHMIC CASE / 53X, 30H(MULTIPLICATIVE ER
     CROR ASSUMED))
C     CALL SUBROUTINE PRINT TO PRINT APPROPRIATE SUBHEADING.
      CALL PRINT
C     ZERO OUT PARAMETERS A, A1, B, C, AND D.  ALSO ZERO OUT
C      VARIANCE-COVARIANCE MATRIX (COV) AND UPPER PORTIONS OF
C      MATRICES G AND H AND COLUMN VECTOR Q.
      A1 = 0.
```

```
      DO 5 I = 1, 4
      F(I) = 0.
      G(I,1) = 0.
      Q(I,1) = 0.
      DO 4 J = 1, 4
      H(I,J) = 0.
      COV(I,J) = 0.
    4 CONTINUE
    5 CONTINUE
   10 DO 13 I = 1, N
C     SET ALL VALUES OF EITHER THIRD (Z) OR SECOND AND THIRD (X, Z)
C      INDEPENDENT VARIABLES TO 1 IF NOT BEING CONSIDERED FOR THIS PASS.
      IF (I1 .LT. 3) Z(I) = 1.0
      IF (I1 .LT. 2) X(I) = 1.0
C     LOGARITHMS OF VARIABLES AT EACH POINT
      YL(I) = ALOG(Y(I))
   12 WL(I) = ALOG(W(I))
      XL(I) = ALOG(X(I))
      ZL(I) = ALOG(Z(I))
   13 CONTINUE
C     SUM OF LOGARITHMS FOR EACH VARIABLE
      DO 14 I = 1, N
      G(1,1) = G(1,1) + YL(I)
      G(2,1) = G(2,1) + WL(I)
      G(3,1) = G(3,1) + XL(I)
      G(4,1) = G(4,1) + ZL(I)
   14 CONTINUE
C     MEAN OF LOGARITHMS FOR EACH VARIABLE
      G(1,1) = G(1,1)/AN
      G(2,1) = G(2,1)/AN
      G(3,1) = G(3,1)/AN
      G(4,1) = G(4,1)/AN
C     CALCULATE MATRIX H AND COLUMN VECTOR Q THAT ARE USED TO OBTAIN
C      SOLUTIONS OF PARAMETERS B, C, D.
      DO 15 I = 1, N
      YDIF = YL(I) - G(1,1)
      WDIF = WL(I) - G(2,1)
      XDIF = XL(I) - G(3,1)
      ZDIF = ZL(I) - G(4,1)
      H(1,1) = H(1,1) + (WDIF**2)
      H(1,2) = H(1,2) + (WDIF * XDIF)
      H(1,3) = H(1,3) + (WDIF * ZDIF)
      H(2,2) = H(2,2) + (XDIF**2)
      H(2,3) = H(2,3) + (XDIF * ZDIF)
      H(3,3) = H(3,3) + (ZDIF**2)
      Q(1,1) = Q(1,1) + (WDIF * YDIF)
      Q(2,1) = Q(2,1) + (XDIF * YDIF)
      Q(3,1) = Q(3,1) + (ZDIF * YDIF)
   15 CONTINUE
   16 H(2,1) = H(1,2)
      H(3,1) = H(1,3)
      H(3,2) = H(2,3)
C     INVERT MATRIX H.
      CALL MATINV(H,I1,Q,1,DETERM,PIVOT,INDEX,IPIVOT)
C     CHECK VALUE OF DETERMINANT GENERATED IN MATRIX INVERSION
C      SUBROUTINE.  IF ZERO (IPIVOT(1) = -1), PRINT ERROR MESSAGE.
```

```
      IF (IPIVOT(1) .EQ. (-1)) GO TO 52
C     OBTAIN VALUES OF PARAMETERS B, C, D STORED IN COLUMN VECTOR Q.
      B = Q(1,1)
      C = Q(2,1)
      D = Q(3,1)
C     PARAMETER A FOR MEDIAN CASE
      A = G(1,1) - (B*G(2,1)) - (C*G(3,1)) - (D*G(4,1))
C     CHECK MAGNITUDE OF VALUE OF PARAMETER A. IF GREATER THAN 88,
C      PRINT ERROR MESSAGE.
      IF (A .GT. 88.) GO TO 50
C     CHECK K1 DESIGNATOR. IF K1 IS ZERO, THIS IS FIRST PASS THROUGH
C      PROGRAM.
      IF (K1 .NE. 0) GO TO 60
      SUMY = 0.
      SUMYSQ = 0.
      YMS = 0.
   60 SIGYSQ = 0.
      DO 17 I = 1, N
C     SUM OF INPUT Y VALUES AND SUM OF SQUARES OF INPUT Y VALUES
      IF (K1 .NE. 0) GO TO 61
      SUMY = SUMY + Y(I)
      SUMYSQ = SUMYSQ + (Y(I)**2)
C     AT EACH POINT CHECK MAGNITUDE OF PRODUCT OF EACH PARAMETER (B,C,D)
C      TIMES LOGARITHM OF VALUE OF INDEPENDENT VARIABLE FOR WHICH
C      PARAMETER IS EXPONENT. IF GREATER THAN 88, PRINT ERROR MESSAGE.
   61 FAB = B * WL(I)
      FAC = C * XL(I)
      FAD = D * ZL(I)
      IF (FAB .GT. 88.) GO TO 50
      IF (FAC .GT. 88.) GO TO 50
      IF (FAD .GT. 88.) GO TO 50
C     SUM OF SQUARES OF DEVIATIONS OF LOGARITHMS OF Y
      SIGYSQ = SIGYSQ + ((YL(I) - A - FAB - FAC - FAD)**2)
   17 CONTINUE
C     SIGYSQ DIVIDED BY DEGREES OF FREEDOM
      SIGYSQ = SIGYSQ/RKM
C     VARIANCE-COVARIANCE MATRIX OF PARAMETERS B, C, D
      DO 180 I = 1, 3
      DO 18 J = 1, 3
      COV(I+1,J+1) = SIGYSQ * H(I,J)
   18 CONTINUE
  180 CONTINUE
C     PARAMETER A FOR MEAN CASE
      A1 = A + (0.5 * SIGYSQ)
C     CHECK MAGNITUDE OF VALUE OF PARAMETER A1. IF GREATER THAN 88,
C      PRINT ERROR MESSAGE.
      IF (A1 .GT. 88.) GO TO 50
C     STANDARD DEVIATIONS OF PARAMETERS B, C, D
      SDEVB = SQRT(COV(2,2))
      SDEVC = SQRT(COV(3,3))
      SDEVD = SQRT(COV(4,4))
      SK1 = ((G(2,1)*H(1,1)) + (G(3,1)*H(2,1)) + (G(4,1)*H(3,1)))*G(2,1)
      SK2 = ((G(2,1)*H(1,2)) + (G(3,1)*H(2,2)) + (G(4,1)*H(3,2)))*G(3,1)
      SK3 = ((G(2,1)*H(1,3)) + (G(3,1)*H(2,3)) + (G(4,1)*H(3,3)))*G(4,1)
      VA = (SIGYSQ/AN) + (SIGYSQ * (SK1 + SK2 + SK3))
      VA1 = VA + (0.5 * (SIGYSQ**2)/RKM)
```

```
C       STANDARD DEVIATIONS OF A AND A1
        SDEVA = SQRT(VA)
        SDEVA1 = SQRT(VA1)
        DO 19 I = 2, 4
C       PARAMETER A INCORPORATED IN VARIANCE-COVARIANCE MATRIX
   19   COV(1,I) = -SIGYSQ*((H(I-1,1)*G(2,1))+(H(I-1,2)*G(3,1))+(H(I-1,3)*
       C G(4,1)))
C       DETERMINATION OF STUDENT'S STATISTICAL T-VALUE (T)
   28   GO TO (29, 30, 31, 32), LS
   29   P = 2.575
        XLS = 0.01
        GO TO 33
   30   P = 1.960
        XLS = 0.05
        GO TO 33
   31   P = 1.645
        XLS = 0.1
        GO TO 33
   32   P = 1.282
        XLS = 0.2
   33   S = KM
        VAR = P
        TT(1) = VAR
        CALL GRT(1,TT,IN,0)
   34   T = TT(1)
C       MEAN OF Y
        YMEAN = SUMY/AN
   35   SUMDS1 = 0.
        SUMDS2 = 0.
        TEMP1 = EXP(SIGYSQ)
        TEMP2 = TEMP1 - 1.0
        TEMP3 = SQRT(TEMP1 * TEMP2)
        TEMP4 = SQRT(TEMP2)
   36   EA = EXP(A)
        EA1 = EXP(A1)
        DO 37 I = 1, N
C       SUM OF SQUARES OF INPUT Y ABOUT THE MEAN
        IF (K1 .EQ. 0) YMS = YMS + ((Y(I) - YMEAN)**2)
        AB = (W(I)**B) * (X(I)**C) * (Z(I)**D)
C       COMPUTED VALUES OF Y FOR MEDIAN CASE
        YC(I) = EA * AB
C       COMPUTED VALUES OF Y FOR MEAN CASE
        YC1(I) = EA1 * AB
C       DEVIATIONS OF Y FOR MEDIAN CASE
        YDEV(I) = Y(I) - YC(I)
C       DEVIATIONS OF Y FOR MEAN CASE
        YDEV1(I) = Y(I) - YC1(I)
C       STANDARD ERROR OF THE ESTIMATE VALUES (ADJUSTED) FOR MEDIAN CASE
        YSD(I) = YC(I) * TEMP3
C       STANDARD ERROR OF THE ESTIMATE VALUES (ADJUSTED) FOR MEAN CASE
        YSD1(I) = YC1(I) * TEMP4
C       SUM OF SQUARES OF Y DEVIATIONS FOR MEDIAN CASE
        SUMDS1 = SUMDS1 + (YDEV(I)**2)
C       SUM OF SQUARES OF Y DEVIATIONS FOR MEAN CASE
        SUMDS2 = SUMDS2 + (YDEV1(I)**2)
C       CALCULATIONS OF UPPER (UP) AND LOWER (XLP) PREDICTION LIMITS
```

```
      DIF1 = WL(I) - G(2,1)
      DIF2 = XL(I) - G(3,1)
      DIF3 = ZL(I) - G(4,1)
      XK1 = ((DIF1*H(1,1)) + (DIF2*H(2,1)) + (DIF3*H(3,1))) * DIF1
      XK2 = ((DIF1*H(1,2)) + (DIF2*H(2,2)) + (DIF3*H(3,2))) * DIF2
      XK3 = ((DIF1*H(1,3)) + (DIF2*H(2,3)) + (DIF3*H(3,3))) * DIF3
      XK = T * SQRT(SIGYSQ * (XK1 + XK2 + XK3 + (1./AN) + 1.))
      UP(I) = YC(I) * EXP(XK)
      XLP(I) = YC(I) * EXP(-XK)
   37 CONTINUE
C     STANDARD DEVIATION OF INPUT Y
   39 IF (K1 .EQ. 0) SDEVY = SQRT(YMS/(AN - 1.0))
      CALL OUTPUT
      K1 = 1
      GO TO 59
C     ERROR MESSAGES
   50 WRITE (6, 51) A, A1, B, C, D
   51 FORMAT (1H0/,  5X,  92HONE OR MORE PARAMETERS IS TOO LARGE FOR THE
     C LOGARITHMIC CASE.  THIS RUN HAS BEEN TERMINATED. ///
     C 5X, 15HA(MEDIAN)    =   , F18.8 /
     C 5X, 15HA(MEAN)      =   , F18.8 /
     C 5X, 15HB            =   , F18.8 /
     C 5X, 15HC            =   , F18.8 /
     C 5X, 15HD            =   , F18.8)
      GO TO 58
   52 WRITE (6, 53)
   53 FORMAT (1H0/, 5X,   102HDETERMINANT IN MATRIX INVERSION SUBROUTINE
     C FOR SUBROUTINE CALC IS ZERO.  THIS RUN HAS BEEN TERMINATED.)
      GO TO 58
   54 WRITE (6, 55)
   55 FORMAT (1H0/, 5X,   64HTHERE ARE INSUFFICIENT DATA SETS.  THIS RUN
     C HAS BEEN TERMINATED.)
      GO TO 58
   56 WRITE (6, 57)
   57 FORMAT (1H0/, 3X,         128HNO SOLUTION HAS BEEN OBTAINED FOR THE
     C LOGARITHMIC CASE BECAUSE OF EXCESSIVE FLOATING-POINT TRAPS.  THIS
     C RUN HAS BEEN TERMINATED.)
   58 IERR = 1
C     RESTORE MODIFIED FLOATING-POINT TRAP ROUTINE FPTCTL TO ITS NORMAL
C      CONDITION
   59 CALL FPTCTL(0)
      RETURN
      END
```

```
$IBFTC PRINT
      SUBROUTINE PRINT
      COMMON  A,          B,          C,          D,          A1,
     C        CV,         PDEVM,      R,          RN,         RN1,
     C        SDEVA,      SDEVA1,     SDEVB,      SDEVC,      SDEVD,
     C        SDEVY,      SEY,        SIGYSQ,     SUMDS1,     SUMDS2,
     C        SUMY,       SUMYSQ,     T,          XLS,        YMEAN
      COMMON  YMS,        I1,         IERR,       IEXPO,      IDESIG,
     C        K1,         KM,         LS,         M,          MN,
     C        N,          COV,        G,          H,          ISC,
     C        RSC,        Q,          PDEV,       UP,         XLP,
     C        WL,         XL,         ZL,         YL,         YC
      COMMON  YC1,        YDEV,       YDEV1,      W,          X,
     C        Z,          EM,         YSD,        YSD1,       TITLE
      COMMON  AN,         EA,         EA1,        IEIG,       IMAT
      DIMENSION  COV(4,4), G(50,50), H(50,50), ISC(4), RSC(4), Q(50,1),
     C IDENT(50), PDEV(50), UP(50), XLP(50), WL(50), XL(50), ZL(50),
     C YL(50), YC(50), YC1(50), YDEV(50), YDEV1(50), W(50), X(50),
     C Z(50), Y(50) , W1(50), X1(50), Z1(50), EM(50,5), YSD(50),
     C YSD1(50), TITLE(10), F(1)
      EQUIVALENCE  (F(1), A), (IDENT(1), EM(1,1)), (Y(1), EM(1,2))
      EQUIVALENCE  (W1(1), EM(1,3)), (X1(1), EM(1,4)), (Z1(1), EM(1,5))
C
C     SUBROUTINE FOR PRINTING SUBHEADINGS
C
      IF (IDESIG .EQ. 0) GO TO 1
      IF (Z1(1) .GT. 0.) GO TO 14
      IF (X1(1) .GT. 0.) GO TO 8
    1 IF (Z1(1) .GT. 0.) GO TO (2, 4, 6, 8, 10, 12, 14), M
      IF (X1(1) .GT. 0.) GO TO (2, 2, 2, 2, 2, 4, 4, 4, 8), M
    2 WRITE (6, 3)
    3 FORMAT (1H0/, 10X, 21HY AS A FUNCTION OF X1 //)
      RETURN
    4 WRITE (6, 5)
    5 FORMAT (1H0/, 10X, 21HY AS A FUNCTION OF X2 //)
      RETURN
    6 WRITE (6, 7)
    7 FORMAT (1H0/, 10X, 21HY AS A FUNCTION OF X3 //)
      RETURN
    8 WRITE (6, 9)
    9 FORMAT (1H0/, 10X, 28HY AS A FUNCTION OF X1 AND X2 //)
      RETURN
   10 WRITE (6, 11)
   11 FORMAT (1H0/, 10X, 28HY AS A FUNCTION OF X1 AND X3 //)
      RETURN
   12 WRITE (6, 13)
   13 FORMAT (1H0/, 10X, 28HY AS A FUNCTION OF X2 AND X3 //)
      RETURN
   14 WRITE (6, 15)
   15 FORMAT (1H0/, 10X, 33HY AS A FUNCTION OF X1, X2, AND X3 //)
      RETURN
      END
```

```
$IBFTC AUX
      SUBROUTINE AUX(RT,FRT)
      COMMON  /LABEL/ S, VAR
C
C     SUBROUTINE USED TO SUPPLY A FUNCTION FOR AUXILIARY SUBROUTINE GRT
C
      CON1 = 8.0*S+3.0
      CON2 = SQRT(S)
      TEMP1 = ALOG(1.0+RT**2/S)
      TEMP2 = SQRT(S*TEMP1)
      TEMP3 = (.368*CON1)/(2.0*CON2*TEMP2)
      TEMP4 = EXP(-TEMP3**2)
      TEMP5 = SQRT(1.0-TEMP4)
      TEMP6 = TEMP2- ((2.0*TEMP2)/CON1)*TEMP5
      FRT = TEMP6 - VAR
      RETURN
      END
```

```
$IBFTC EXPO
      SUBROUTINE EXPO
      COMMON  A,          B,          C,          D,          A1,
     C        CV,         PDEVM,      R,          RN,         RN1,
     C        SDEVA,      SDEVA1,     SDEVB,      SDEVC,      SDEVD,
     C        SDEVY,      SEY,        SIGYSQ,     SUMDS1,     SUMDS2,
     C        SUMY,       SUMYSQ,     T,          XLS,        YMEAN
      COMMON  YMS,        I1,         IERR,       IEXPO,      IDESIG,
     C        K1,         KM,         LS,         M,          MN,
     C        N,          COV,        G,          H,          ISC,
     C        RSC,        Q,          PDEV,       UP,         XLP,
     C        WL,         XL,         ZL,         YL,         YC
      COMMON  YC1,        YDEV,       YDEV1,      W,          X,
     C        Z,          EM,         YSD,        YSD1,       TITLE
      COMMON  AN,         EA,         EA1,        IEIG,       IMAT
      DIMENSION  COV(4,4), G(50,50), H(50,50), ISC(4), RSC(4), Q(50,1),
     C IDENT(50), PDEV(50), UP(50), XLP(50), WL(50), XL(50), ZL(50),
     C YL(50), YC(50), YC1(50), YDEV(50), YDEV1(50), W(50), X(50),
     C Z(50), Y(50) , W1(50), X1(50), Z1(50), EM(50,5), YSD(50),
     C YSD1(50), TITLE(10), F(1)
      DIMENSION  Q1(11), ATEMP(11), BTEMP(11), CTEMP(11), DTEMP(11),
     C PIVOT(50), IPIVOT(50), INDEX(50,2)
      EQUIVALENCE  (F(1), A), (IDENT(1), EM(1,1)), (Y(1), EM(1,2))
      EQUIVALENCE  (W1(1), EM(1,3)), (X1(1), EM(1,4)), (Z1(1), EM(1,5))
C
C     SUBROUTINE FOR OBTAINING SOLUTIONS FOR EXPONENTIAL CASE
C
    1 ASSIGN 160 TO III
      CALL FPTCTL(III)
C     PRINT TITLE AND HEADING.
      WRITE (6, 2) (TITLE(I), I = 1, 10)
    2 FORMAT (1H1, 9X, 9A6, A4 /// 60X, 16HEXPONENTIAL CASE / 56X,
     C 24H(ADDITIVE ERROR ASSUMED))
C     CALL SUBROUTINE PRINT TO PRINT APPROPRIATE SUBHEADING.
      CALL PRINT
C     SET SOLUTION DESIGNATOR TO ZERO.
      K2 = 0
    3 DO 15 L = 1, 50
C     CLEAR UPPER PORTIONS OF MATRIX G AND COLUMN VECTOR Q.
      DO 5 I = 1, 4
      Q(I,1) = 0.
      DO 4 J = 1, 4
      G(I,J) = 0.
    4 CONTINUE
    5 CONTINUE
C     CHECK MAGNITUDE OF VALUE OF PARAMETER A. IF GREATER THAN 88,
C      PRINT ERROR MESSAGE.
      IF (A .GT. 88.) GO TO 17
      EA = EXP(A)
      DO 6 I = 1, N
      FAB = B * WL(I)
      FAC = C * XL(I)
      FAD = D * ZL(I)
C     AT EACH POINT CHECK MAGNITUDE OF PRODUCT OF EACH PARAMETER (B,C,D)
C      TIMES LOGARITHM OF VALUE OF INDEPENDENT VARIABLE FOR WHICH
C      PARAMETER IS EXPONENT. IF GREATER THAN 88, PRINT ERROR MESSAGE.
```

```
      IF (FAB .GT. 88.) GO TO 17
      IF (FAC .GT. 88.) GO TO 17
      IF (FAD .GT. 88.) GO TO 17
C     COMPUTED VALUES OF Y FOR EXPONENTIAL CASE
      YC(I) = EA * (W(I)**B) * (X(I)**C) * (Z(I)**D)
      YA = YC(I)**2
C     DEVIATIONS OF Y FOR EXPONENTIAL CASE
      YDEV(I) = Y(I) - YC(I)
C     IF K2 IS EQUAL TO 1, SIGNIFYING THAT A SOLUTION HAS BEEN
C      OBTAINED, SKIP CALCULATIONS OF MATRIX G AND COLUMN VECTOR Q.
      IF (K2 .EQ. 1) GO TO 6
      YB = YDEV(I) * YC(I)
C     CALCULATE MATRIX G AND COLUMN VECTOR Q THAT ARE USED TO OBTAIN
C      SOLUTION OF CORRECTIONS (DA, DB, DC, DD) TO PARAMETERS A, B, C, D
      G(1,1) = G(1,1) + YA
      G(1,2) = G(1,2) + (YA * WL(I))
      G(1,3) = G(1,3) + (YA * XL(I))
      G(1,4) = G(1,4) + (YA * ZL(I))
      G(2,2) = G(2,2) + (YA * (WL(I)**2))
      G(2,3) = G(2,3) + (YA * WL(I) * XL(I))
      G(2,4) = G(2,4) + (YA * WL(I) * ZL(I))
      G(3,3) = G(3,3) + (YA * (XL(I)**2))
      G(3,4) = G(3,4) + (YA * XL(I) * ZL(I))
      G(4,4) = G(4,4) + (YA * (ZL(I)**2))
      Q(1,1) = Q(1,1) + YB
      Q(2,1) = Q(2,1) + (YB * WL(I))
      Q(3,1) = Q(3,1) + (YB * XL(I))
      Q(4,1) = Q(4,1) + (YB * ZL(I))
    6 CONTINUE
      IF (K2 .EQ. 1) GO TO 22
      G(2,1) = G(1,2)
      G(3,1) = G(1,3)
      G(3,2) = G(2,3)
      G(4,1) = G(1,4)
      G(4,2) = G(2,4)
      G(4,3) = G(3,4)
C     INVERT MATRIX G.
      CALL MATINV(G,MN,Q,1,DETERM,PIVOT,INDEX,IPIVOT)
C     CHECK VALUE OF DETERMINANT GENERATED IN MATRIX INVERSION
C      SUBROUTINE. IF ZERO (IPIVOT(1) = -1), PRINT ERROR MESSAGE.
      IF (IPIVOT(1) .EQ. (-1)) GO TO 19
C     OBTAIN VALUES OF DA, DB, DC, DD STORED IN COLUMN VECTOR Q.
      DA = Q(1,1)
      DB = Q(2,1)
      DC = Q(3,1)
      DD = Q(4,1)
      TEMP = 1.0
    7 TEMP = 0.1 * TEMP
      DO 8 J = 1, 11
      FI = TEMP * FLOAT(J - 1)
C     ADD FRACTIONAL PARTS OF CORRECTIONS TO EACH PARAMETER (INITIALLY
C      FROM 0. TO 1. IN STEPS OF 0.1). ELEVEN SETS OF PARAMETERS ARE
C      THUS OBTAINED.
      ATEMP(J) = A + (DA * FI)
      BTEMP(J) = B + (DB * FI)
      CTEMP(J) = C + (DC * FI)
```

```
      DTEMP(J) = D + (DD * FI)
      Q1(J) = 0.
    8 CONTINUE
C     CALCULATE SUM OF SQUARES OF Y DEVIATIONS FOR EACH SET OF
C      PARAMETERS.
      DO 90 J = 1, 11
      ETEMP = EXP(ATEMP(J))
      DO 9 I = 1, N
      YTEMP = ETEMP*(W(I)**BTEMP(J))*(X(I)**CTEMP(J))*(Z(I)**DTEMP(J))
      Q1(J) = Q1(J) + ((Y(I) - YTEMP)**2)
    9 CONTINUE
   90 CONTINUE
C     FIND WHICH SET (LM) OF PARAMETERS GIVES LOWEST SUM OF SQUARES OF
C      Y DEVIATIONS.
      SUMDS1 = Q1(1)
      LM = 1
      DO 10 J = 2, 11
      IF (SUMDS1 .LE. Q1(J)) GO TO 10
      LM = J
      SUMDS1 = Q1(J)
   10 CONTINUE
C     IF LM IS 1 AND IF CORRECTIONS ARE GREATER THAN 0.00000001, REPEAT
C      PROCEDURE USING FRACTIONAL PARTS FROM 0. TO 0.1 IN STEPS OF 0.01,
C      ETC.
      IF (LM .GT. 1) GO TO 11
      IF (ABS(DA*10.**TEMP) .GT. 1.0E-8) GO TO 7
      IF (ABS(DB*10.**TEMP) .GT. 1.0E-8) GO TO 7
      IF (ABS(DC*10.**TEMP) .GT. 1.0E-8) GO TO 7
      IF (ABS(DD*10.**TEMP) .GT. 1.0E-8) GO TO 7
C     ABSOLUTE DIFFERENCE BETWEEN PREVIOUS AND PRESENT VALUE OF EACH
C      PARAMETER (DDA, DDB, DDC, DDD)
   11 DDA = ABS(A - ATEMP(LM))
      DDB = ABS(B - BTEMP(LM))
      DDC = ABS(C - CTEMP(LM))
      DDD = ABS(D - DTEMP(LM))
C     STORE PRESENT VALUE OF EACH PARAMETER.
      A = ATEMP(LM)
      B = BTEMP(LM)
      C = CTEMP(LM)
      D = DTEMP(LM)
C     IF DDA, DDB, DDC, DDD ARE EACH EQUAL TO OR LESS THAN 0.00000001, A
C      SOLUTION IS ASSUMED. SET K2 EQUAL TO 1.
   13 IF (DDA .GT. 1.0E-8) GO TO 15
      IF (DDB .GT. 1.0E-8) GO TO 15
      IF (DDC .GT. 1.0E-8) GO TO 15
      IF (DDD .LE. 1.0E-8) K2 = 1
   15 CONTINUE
C     ERROR MESSAGES
      WRITE (6, 16)
   16 FORMAT (1H0/, 5X,          86HNO SOLUTION HAS BEEN OBTAINED FOR THE
     C EXPONENTIAL CASE.  THIS RUN HAS BEEN TERMINATED.)
      GO TO 21
  160 WRITE (6, 165)
  165 FORMAT (1H0/, 3X,          128HNO SOLUTION HAS BEEN OBTAINED FOR THE
     C EXPONENTIAL CASE BECAUSE OF EXCESSIVE FLOATING-POINT TRAPS.  THIS
     C RUN HAS BEEN TERMINATED.)
```

```
   GO TO 21
17 WRITE (6, 18) A, B, C, D
18 FORMAT (1HO/,  5X,  92HONE OR MORE PARAMETERS IS TOO LARGE FOR THE
  C EXPONENTIAL CASE.  THIS RUN HAS BEEN TERMINATED. ///
  C 5X, 15HA            =   , F18.8 /
  C 5X, 15HB            =   , F18.8 /
  C 5X, 15HC            =   , F18.8 /
  C 5X, 15HD            =   , F18.8)
   GO TO 21
19 WRITE (6, 20)
20 FORMAT (1HO/, 5X,   102HDETERMINANT IN MATRIX INVERSION SUBROUTINE
  C FOR SUBROUTINE EXPO IS ZERO.  THIS RUN HAS BEEN TERMINATED.)
21 IERR = 1
22 CALL FPTCTL(0)
   RETURN
   END
```

```
$IBFTC STATIS
      SUBROUTINE STATIS
      COMMON  A,          B,          C,          D,          A1,
     C        CV,         PDEVM,      R,          RN,         RN1,
     C        SDEVA,      SDEVA1,     SDEVB,      SDEVC,      SDEVD,
     C        SDEVY,      SEY,        SIGYSQ,     SUMDS1,     SUMDS2,
     C        SUMY,       SUMYSQ,     T,          XLS,        YMEAN
      COMMON  YMS,        I1,         IERR,       IEXPO,      IDESIG,
     C        K1,         KM,         LS,         M,          MN,
     C        N,          COV,        G,          H,          ISC,
     C        RSC,        Q,          PDEV,       UP,         XLP,
     C        WL,         XL,         ZL,         YL,         YC
      COMMON  YC1,        YDEV,       YDEV1,      W,          X,
     C        Z,          EM,         YSD,        YSD1,       TITLE
      COMMON  AN,         EA,         EA1,        IEIG,       IMAT
      DIMENSION  COV(4,4), G(50,50), H(50,50), ISC(4), RSC(4), Q(50,1),
     C IDENT(50), PDEV(50), UP(50), XLP(50), WL(50), XL(50), ZL(50),
     C YL(50), YC(50), YC1(50), YDEV(50), YDEV1(50), W(50), X(50),
     C Z(50), Y(50) , W1(50), X1(50), Z1(50), EM(50,5), YSD(50),
     C YSD1(50), TITLE(10), F(1)
      DIMENSION  V(50,50), PIVOT(50), IPIVOT(50), INDEX(50,2), VALU(50),
     C DIAG(50), SUPERD(50), Q2(49), VALL(50), S1(49), D1(50), C1(49),
     C U(50), IND1(50)
      EQUIVALENCE  (F(1), A), (IDENT(1), EM(1,1)), (Y(1), EM(1,2))
      EQUIVALENCE  (W1(1), EM(1,3)), (X1(1), EM(1,4)), (Z1(1), EM(1,5))
C
C     SUBROUTINE FOR CALCULATING STATISTICS FOR EXPONENTIAL CASE
C
    1 ASSIGN 27 TO III
      CALL FPTCTL(III)
C     SET ERROR DESIGNATORS TO ZERO.
      IEIG = 0
      IMAT = 0
      IFP = 0
C     CLEAR MATRICES G AND H, COLUMN VECTOR Q, VARIABLES UP AND XLP, AND
C     VARIANCE-COVARIANCE MATRIX (COV).
      DO 3 I = 1, 50
      IF (I .LT. 6) F(I+10) = 0.
      IF (I .LT. 17) F(I+36) = 0.
      Q(I,1) = 0.
      UP(I) = 0.
      XLP(I) = 0.
      DO 2 J = 1, 50
      G(I,J) = 0.
      H(I,J) = 0.
      V(I,J) = 0.
    2 CONTINUE
    3 CONTINUE
      MN1 = MN + 1
C     CASE DESIGNATOR (LOGARITHMIC -- IEXPO = 0,
C      EXPONENTIAL -- IEXPO = 1)
    4 IEXPO = 1
      PDEVM = 0.
    5 DO 7 I = 1, N
C     PERCENT DEVIATIONS OF Y
    6 PDEV(I) = (YDEV(I)/Y(I)) * 100.0
```

```
C       SUM OF ABSOLUTE PERCENT DEVIATIONS
        PDEVM = PDEVM + ABS(PDEV(I))
      7 CONTINUE
        SIGYSQ = SUMDS1/FLOAT(KM)
C       STANDARD ERROR OF THE ESTIMATE (ADJUSTED)
        SEY = SQRT(SIGYSQ)
C       COEFFICIENT OF VARIATION (PERCENT)
      8 CV = (SEY/YMEAN) * 100.0
C       COEFFICIENT OF CORRELATION (UNADJUSTED)
        R = SQRT(1.0 - (SUMDS1/YMS))
C       AVERAGE ABSOLUTE PERCENT DEVIATION
        PDEVM = PDEVM/AN
C       CALCULATIONS FROM THIS POINT THROUGH STATEMENT NUMBER 25 ARE
C        FOR THE PURPOSE OF OBTAINING THE VARIANCE-COVARIANCE MATRIX OF
C        THE PARAMETERS A, B, C, D AS DESCRIBED IN RAND MEMORANDUM
C        4879-PR (1967), 'MULTIVARIATE LOGARITHMIC AND EXPONENTIAL
C        REGRESSION MODELS', BY C.A. GRAVER AND H.E. BOREN, JR.
        DO 10 I = 1, N
        G(I,1) = 1.0
        G(I,2) = WL(I)
        G(I,3) = XL(I)
        G(I,4) = ZL(I)
     10 CONTINUE
        DO 211 I = 1, MN
        DO 111 J = 1, MN
        DO 11 K = 1, N
        H(I,J) = H(I,J) + (G(K,I) * G(K,J))
     11 CONTINUE
    111 CONTINUE
    211 CONTINUE
        CALL MATINV(H,MN,Q,0,DETERM,PIVOT,INDEX,IPIVOT)
C       CHECK VALUE OF DETERMINANT GENERATED IN MATRIX INVERSION
C        SUBROUTINE. IF ZERO (IPIVOT(1) = -1), PRINT ERROR MESSAGE.
        IF (IPIVOT(1) .EQ. (-1)) IMAT = 1
        IF (IMAT .EQ. 1) GO TO 28
        DO 212 I = 1, MN
        DO 112 J = 1, N
        DO 12 K = 1, MN
        V(I,J) = V(I,J) + (H(I,K) * G(J,K))
     12 CONTINUE
    112 CONTINUE
    212 CONTINUE
        DO 113 I = 1, N
        DO 13 J = 1, N
        H(I,J) = 0.
     13 CONTINUE
    113 CONTINUE
        DO 214 I = 1, N
        DO 114 J = 1, N
        DO 14 K = 1, MN
        H(I,J) = H(I,J) - (G(I,K) * V(K,J))
     14 CONTINUE
    114 CONTINUE
    214 CONTINUE
        DO 16 I = 1, N
        H(I,I) = 1.0 + H(I,I)
```

```
   16 CONTINUE
      CALL EIGEN(H,VALU,N,1,G,DIAG,SUPERD,Q2,VALL,S1,D1,C1,U,IND1,NOEIG)
C     CHECK FOR ERROR IN SUBROUTINE EIGEN (NOEIG = 1).
      IF (NOEIG .EQ. 1) IEIG = 1
      IF (IEIG .EQ. 1) GO TO 28
      DO 117 I = 1, KM
      DO 17 J = 1, N
      G(I,J) = H(J,I)/YC(J)
   17 CONTINUE
  117 CONTINUE
      DO 118 I = 1, MN
      DO 18 J = 1, N
      H(I,J) = V(I,J)/YC(J)
      V(I,J) = 0.
   18 CONTINUE
  118 CONTINUE
      DO 219 I = 1, KM
      DO 119 J = 1, KM
      DO 19 K = 1, N
      V(I,J) = V(I,J) + (G(I,K) * G(J,K))
   19 CONTINUE
  119 CONTINUE
  219 CONTINUE
      CALL MATINV(V,KM,Q,0,DETERM,PIVOT,INDEX,IPIVOT)
C     CHECK VALUE OF DETERMINANT GENERATED IN MATRIX INVERSION
C      SUBROUTINE. IF ZERO (IPIVOT(1) = -1), PRINT ERROR MESSAGE.
      IF (IPIVOT(1) .EQ. (-1)) IMAT = 1
      IF (IMAT .EQ. 1) GO TO 28
      DO 221 I = 1, KM
      DO 20 J = 1, KM
      UP(J) = V(I,J)
   20 CONTINUE
      DO 121 K = 1, N
      V(I,K) = 0.
      DO 21 L = 1, KM
      V(I,K) = V(I,K) + (UP(L) * G(L,K))
   21 CONTINUE
  121 CONTINUE
  221 CONTINUE
      DO 223 I = 1, N
      DO 22 J = 1, KM
      UP(J) = V(J,I)
   22 CONTINUE
      DO 123 K = 1, N
      V(K,I) = 0.
      DO 23 L = 1, N
      V(K,I) = V(K,I) - (UP(L) * G(L,K))
   23 CONTINUE
  123 CONTINUE
  223 CONTINUE
      DO 24 I = 1, N
      UP(I) = 0.
      V(I,I) = 1.0 + V(I,I)
   24 CONTINUE
      CALL EIGEN(V,VALU,N,1,G,DIAG,SUPERD,Q2,VALL,S1,D1,C1,U,IND1,NOEIG)
C     CHECK FOR ERROR IN SUBROUTINE EIGEN (NOEIG = 1).
      IF (NOEIG .EQ. 1) IEIG = 1
```

```
      IF (IEIG .EQ. 1) GO TO 28
      DO 324 I = 1, N
      DO 224 J = 1, N
      G(I,J) = 0.
      DO 124 K = 1, MN
      G(I,J) = G(I,J) + (V(I,K) * VALU(K) * V(J,K))
  124 CONTINUE
  224 CONTINUE
  324 CONTINUE
      DO 425 I = 1, MN
      DO 325 J = 1, MN
      V(I,J) = 0.
      DO 225 K = 1, N
      DO 125 L = 1, N
      V(I,J) = V(I,J) + (H(I,K) * G(K,L) * H(J,L))
  125 CONTINUE
  225 CONTINUE
  325 CONTINUE
  425 CONTINUE
      DO 550 I = 1, MN
      DO 25 J = 1, MN
      COV(I,J) = SIGYSQ * V(I,J)
   25 CONTINUE
  550 CONTINUE
C     STANDARD DEVIATIONS OF PARAMETERS A, B, C, D
   26 SDEVA = SQRT(COV(1,1))
      SDEVB = SQRT(COV(2,2))
      SDEVC = SQRT(COV(3,3))
      SDEVD = SQRT(COV(4,4))
      GO TO 28
   27 IFP = 1
   28 DO 30 I = 1, N
      IF (IEIG .EQ. 1 .OR. IMAT .EQ. 1 .OR. IFP .EQ. 1) GO TO 29
C     CALCULATIONS OF UPPER (UP) AND LOWER (XLP) PREDICTION LIMITS
      XK =  V(1,1) + (V(2,2) * (WL(I)**2)) + (V(3,3) * (XL(I)**2))
     C + (V(4,4) * (ZL(I)**2)) + (2. * WL(I) * V(1,2)) + (2. * XL(I) *
     C V(1,3)) + (2. * ZL(I) * V(1,4)) + (2. * WL(I) * XL(I) * V(2,3))
     C + (2. * WL(I) * ZL(I) * V(2,4)) + (2. * XL(I) * ZL(I) * V(3,4))
      XK = T * SQRT(SIGYSQ * (1.0 + ((YC(I)**2) * XK)))
      UP(I) = YC(I) + XK
      XLP(I) = YC(I) - XK
C     TWICE COMPUTED Y VALUE MINUS INPUT Y VALUE AT EACH POINT. IF
C      THIS CALCULATION IS POSITIVE FOR ALL POINTS, THEN MATRIX OF
C      SECOND PARTIAL DERIVATIVES IS POSITIVE DEFINITE AND SOLUTIONS
C      OF PARAMETERS A, B, C, D GIVE ABSOLUTE MINIMIZATION OF THE SUM
C      OF SQUARES OF Y DEVIATIONS IN THE RANGE OF DATA USED FOR THIS
C      RUN.
   29 YDEV1(I) = (2.0 * YC(I)) - Y(I)
   30 CONTINUE
C     ERROR MESSAGES
   31 IF (IEIG .EQ. 0) GO TO 33
      WRITE (6, 32)
   32 FORMAT (1HO/, 5X,    122HREQUIRED NUMBER OF EIGENVECTORS WERE NOT
     COBTAINED.  THE PARAMETER VARIANCES AND T-RELATED STATISTICS CANN(
     C BE CALCULATED.)
      GO TO 37
```

```
33 IF (IMAT .EQ. 0) GO TO 35
   WRITE (6, 34)
34 FORMAT (1HO/, 5X,   123HDETERMINANT IN MATRIX INVERSION SUBROUTINE
  C IS ZERO.  THE PARAMETER VARIANCES AND T-RELATED STATISTICS CANNOT
  C BE CALCULATED.)
   GO TO 37
35 IF (IFP .EQ. 0) GO TO 37
   WRITE (6, 36)
36 FORMAT (1HO/, 1X,      130HEXCESSIVE FLOATING-POINT TRAPS EXIST IN
  C SUBROUTINE STATIS.  THE PARAMETER VARIANCES AND T-RELATED STATIST
  CICS CANNOT BE CALCULATED.)
   IMAT = 1
37 CALL FPTCTL(0)
   CALL OUTPUT
   RETURN
   END
```

```
$IBFTC OUTPUT
      SUBROUTINE OUTPUT
      COMMON  A,          B,          C,          D,          A1,
     C        CV,         PDEVM,      R,          RN,         RN1,
     C        SDEVA,      SDEVA1,     SDEVB,      SDEVC,      SDEVD,
     C        SDEVY,      SEY,        SIGYSQ,     SUMDS1,     SUMDS2,
     C        SUMY,       SUMYSQ,     T,          XLS,        YMEAN
      COMMON  YMS,        I1,         IERR,       IEXPO,      IDESIG,
     C        K1,         KM,         LS,         M,          MN,
     C        N,          COV,        G,          H,          ISC,
     C        RSC,        Q,          PDEV,       UP,         XLP,
     C        WL,         XL,         ZL,         YL,         YC
      COMMON  YC1,        YDEV,       YDEV1,      W,          X,
     C        Z,          EM,         YSD,        YSD1,       TITLE
      COMMON  AN,         EA,         EA1,        IEIG,       IMAT
      DIMENSION  COV(4,4), G(50,50), H(50,50), ISC(4), RSC(4), Q(50,1),
     C IDENT(50), PDEV(50), UP(50), XLP(50), WL(50), XL(50), ZL(50),
     C YL(50), YC(50), YC1(50), YDEV(50), YDEV1(50), W(50), X(50),
     C Z(50), Y(50) , W1(50), X1(50), Z1(50), EM(50,5), YSD(50),
     C YSD1(50), TITLE(10), F(1)
      EQUIVALENCE  (F(1), A), (IDENT(1), EM(1,1)), (Y(1), EM(1,2))
      EQUIVALENCE  (W1(1), EM(1,3)), (X1(1), EM(1,4)), (Z1(1), EM(1,5))
C
C     SUBROUTINE FOR PRINTING OUTPUT FOR BOTH LOGARITHMIC AND
C      EXPONENTIAL CASES
C
C
C     CHECK TO DETERMINE WHICH CASE IS BEING CONSIDERED
C      (LOGARITHMIC -- IEXPO = 0, EXPONENTIAL -- IEXPO = 1).
    1 IF (IEXPO .EQ. 0) GO TO 10
    2 WRITE (6, 3) A, B, C, D, PDEVM, YMEAN, SEY, CV, R
    3 FORMAT (1H0//, 89X,10HSTD. ERROR, 5X, 9HCOEFF. OF, 5X,9HCOEFF. OF/
     C 62X, 9HAVG. PCT., 18X, 11HOF ESTIMATE, 4X, 9HVARIATION, 4X,
     C 11HCORRELATION / 10X, 1HA, 13X, 1HB, 13X, 1HC, 13X, 1HD, 9X,
     C 9HDEV. OF Y, 6X, 6HY MEAN, 6X, 10H(ADJUSTED), 5X, 9H(PERCENT),
     C 4X, 12H(UNADJUSTED) // 2X, 9(F12.5, 2X))
      WRITE (6, 4) SUMDS1, SDEVY, SDEVA, SDEVB, SDEVC, SDEVD, COV(1,2),
     C COV(1,3), COV(1,4)
    4 FORMAT (1H0//, 20X,9HSTD. DEV. / 5X, 10HSUM OF SQ., 4X,10HOF INPUT
     C Y, 5X, 4(9HSTD. DEV., 5X), 1X, 2(7HCOV. OF, 7X), 7HCOV. OF / 6X,
     C 7HOF YDEV, 6X, 10H(ADJUSTED), 7X, 4HOF A, 10X, 4HOF B, 10X,
     C 4HOF C, 10X, 4HOF D, 11X, 3HA,B, 11X, 3HA,C, 11X, 3HA,D // 2X,
     C 9(F12.5, 2X))
      WRITE (6, 5) COV(2,3), COV(2,4), COV(3,4), EA
    5 FORMAT (1H0//, 7X, 3(7HCOV. OF, 7X), 6HEXP(A) / 9X, 3HB,C, 11X,
     C 3HB,D, 11X, 3HC,D // 2X, 4(F12.5, 2X))
      GO TO 37
   52 WRITE (6, 7) XLS
    7 FORMAT (1H1, 14X, 22H. . . INPUT DATA . . ., 6X, 14(1H., 1X),
     C 16HCALCULATED DATA , 14(1H., 1X) /// 96X, 5HLOWER, 8X, 5HUPPER /
     C 95X, 2(8HPREDICT.,5X) / 58X, 4HYDEV, 8X, 6HYDEV/Y, 20X, 2(5HLIMIT
     C , 8X) / 14X, 5HLABEL, 14X, 1HY, 10X, 7HY CALC.,4X, 10H(Y-YCALC.),
     C 3X, 9H(PERCENT), 4X, 11H2YCALC. - Y, 1X, 20H(LEVEL OF SIGNIF. = ,
     C F4.2, 1H) /)
      NOTE = 0
      DO 9 I = 1, N
```

```
      IF (YDEV1(I) .LE. 0.) NOTE = 1
      WRITE (6, 8) IDENT(I), Y(I), YC(I), YDEV(I), PDEV(I), YDEV1(I),
     C XLP(I), UP(I)
    8 FORMAT (1H , 13X, A6, 5X, 7(F12.5, 1X))
    9 CONTINUE
      IF (NOTE .EQ. 1) WRITE (6, 90)
   90 FORMAT (1HO , 108HTHE ABOVE SOLUTION FOR THIS RUN DOES NOT GIVE AN
     C ABSOLUTE MINIMIZATION OF THE SUM OF SQUARES OF Y RESIDUALS.)
      NOTE = 0
      RETURN
   10 WRITE (6, 11) A, A1, B, C, D, YMEAN
   11 FORMAT (1HO//, 6X, 9HA(MEDIAN), 6X, 7HA(MEAN), 10X, 1HB, 13X, 1HC,
     C 13X, 1HD, 10X, 6HY MEAN // 2X, 6(F12.5, 2X))
      WRITE (6, 12) SUMDS1, SUMDS2, SDEVY, SDEVA, SDEVA1, SDEVB, SDEVC,
     C SDEVD, COV(1,2)
   12 FORMAT (1HO//, 5X, 2(10HSUM OF SQ., 4X), 1X, 9HSTD. DEV. / 6X,
     C 7HOF YDEV, 7X, 7HOF YDEV, 6X, 10HOF INPUT Y, 5X, 5(9HSTD. DEV.,
     C 5X), 1X, 7HCOV. OF / 6X, 8H(MEDIAN), 7X, 6H(MEAN), 6X, 10H(ADJUST
     CED), 4X, 12HOF A(MEDIAN), 3X, 10HOF A(MEAN), 6X, 4HOF B, 10X,
     C 4HOF C, 10X, 4HOF D, 11X, 3HA,B // 2X, 9(F12.5, 2X))
      WRITE (6, 31) COV(1,3), COV(1,4), COV(2,3), COV(2,4), COV(3,4),
     C EA, EA1
   31 FORMAT (1HO//, 7X, 5(7HCOV. OF, 7X), 2(6HEXP(A), 8X) / 9X,
     C 3HA,C, 11X, 3HA,D, 11X, 3HB,C, 11X, 3HB,D, 11X, 3HC,D, 8X,
     C 8H(MEDIAN), 7X, 6H(MEAN) // 2X, 7(F12.5, 2X))
   37 WRITE (6, 38) T
   38 FORMAT (1HO////, 7X, 12HT VALUE   = , F12.5)
      IF (IEIG .EQ. 1 .OR. IMAT .EQ. 1) GO TO 51
      AT = ABS(A/SDEVA)
      IF (IEXPO .EQ. 1) GO TO 39
      AT1 = ABS(A1/SDEVA1)
      IF (AT .GE. T) WRITE (6, 41)
      IF (AT .LT. T) WRITE (6, 42)
      IF (AT1 .GE. T) WRITE (6, 142)
      IF (AT1 .LT. T) WRITE (6, 143)
   39 IF (IEXPO .EQ. 0) GO TO 40
      IF (AT .GE. T) WRITE (6, 43)
      IF (AT .LT. T) WRITE (6, 44)
   40 BT = ABS(B/SDEVB)
      IF (BT .GE. T) WRITE (6, 45)
      IF (BT .LT. T) WRITE (6, 46)
      IF (I1 .EQ. 1) GO TO 51
      CT = ABS(C/SDEVC)
      IF (CT .GE. T) WRITE (6, 47)
      IF (CT .LT. T) WRITE (6, 48)
      IF (I1 .EQ. 2) GO TO 51
      DT = ABS(D/SDEVD)
      IF (DT .GE. T) WRITE (6, 49)
      IF (DT .LT. T) WRITE (6, 50)
   51 IF (IEXPO .EQ. 1) GO TO 52
   41 FORMAT (1HO, 6X, 30HA (MEDIAN CASE) IS SIGNIFICANT)
   42 FORMAT (1HO, 6X, 34HA (MEDIAN CASE) IS NOT SIGNIFICANT)
  142 FORMAT (1HO, 6X, 28HA (MEAN CASE) IS SIGNIFICANT)
  143 FORMAT (1HO, 6X, 32HA (MEAN CASE) IS NOT SIGNIFICANT)
   43 FORMAT (1HO, 6X, 16HA IS SIGNIFICANT)
   44 FORMAT (1HO, 6X, 20HA IS NOT SIGNIFICANT)
```

```
45 FORMAT (1HO, 6X, 16HB IS SIGNIFICANT)
46 FORMAT (1HO, 6X, 20HB IS NOT SIGNIFICANT)
47 FORMAT (1HO, 6X, 16HC IS SIGNIFICANT)
48 FORMAT (1HO, 6X, 20HC IS NOT SIGNIFICANT)
49 FORMAT (1HO, 6X, 16HD IS SIGNIFICANT)
50 FORMAT (1HO, 6X, 20HD IS NOT SIGNIFICANT)
   KOUNT = 0
   WRITE (6, 13) XLS
13 FORMAT (1H1, 2X, 10(1H., 1X), 11HINPUT DATA , 10(1H., 1X), 7X,
  C 9(1H., 1X), 32HCALCULATED DATA FOR MEDIAN CASE , 9(1H., 1X) ///
  C 109X, 5HLOWER, 7X, 5HUPPER / 82X, 10HSTD. ERROR, 3X, 9HCOEFF. OF,
  C 4X, 8HPREDICT., 4X,8HPREDICT./ 74X, 4HYDEV, 4X, 11HOF ESTIMATE,
  C 2X, 9HVARIATION, 5X, 5HLIMIT, 7X, 5HLIMIT / 1X, 5HLABEL, 9X, 1HY,
  C 10X, 2HX1, 10X, 2HX2, 10X, 2HX3, 9X, 7HY CALC.,3X, 10H(Y-YCALC.),
  C 1X, 10H(ADJUSTED), 3X, 9H(PERCENT), 1X, 20H(LEVEL OF SIGNIF. = ,
  C F4.2, 1H) /)
60 DO 28 I = 1, N
   IF (KOUNT .NE. 0) GO TO 65
   YCA = YC(I)
   YDA = YDEV(I)
   YSDA = YSD(I)
   CV = (YSD(I)/YMEAN) * 100.
   GO TO 70
65 YCA = YC1(I)
   YDA = YDEV1(I)
   YSDA = YSD1(I)
   CV = (YSD1(I)/YMEAN) * 100.
70 IF (IDESIG .EQ. 0) GO TO 30
   IF (IDESIG .EQ. 0) GO TO 30
   IF (Z1(1) .GT. 0.) GO TO 26
   IF (X1(1) .GT. 0.) GO TO 20
30 IF (Z1(1) .GT. 0.) GO TO (14, 16, 18, 20, 22, 24, 26), M
   IF (X1(1) .GT. 0.) GO TO (14, 14, 14, 14, 14, 16, 16, 16, 20), M
14 WRITE (6, 15) IDENT(I), Y(I), W1(I), YCA, YDA, YSDA, CV, XLP(I),
  C UP(I)
15 FORMAT (1H , A6, 1X, 2(F11.3, 1X), 24X, 2(F11.3, 1X),2(F11.5,1X),
  C 2(F11.3, 1X))
   GO TO 28
16 WRITE (6, 17) IDENT(I), Y(I), X1(I), YCA, YDA, YSDA, CV, XLP(I),
  C UP(I)
17 FORMAT (1H , A6, 1X, 2(F11.3, 13X), 2(F11.3, 1X), 2(F11.5, 1X),
  C 2(F11.3, 1X))
   GO TO 28
18 WRITE (6, 19) IDENT(I), Y(I), Z1(I), YCA, YDA, YSDA, CV, XLP(I),
  C UP(I)
19 FORMAT (1H , A6, 1X, F11.3, 25X, 3(F11.3, 1X), 2(F11.5, 1X),
  C 2(F11.3, 1X))
   GO TO 28
20 WRITE (6, 21) IDENT(I), Y(I), W1(I), X1(I), YCA, YDA, YSDA, CV,
  C XLP(I), UP(I)
21 FORMAT (1H , A6, 1X, 3(F11.3, 1X), 12X, 2(F11.3, 1X), 2(F11.5,1X),
  C 2(F11.3, 1X))
   GO TO 28
22 WRITE (6, 23) IDENT(I), Y(I), W1(I), Z1(I), YCA, YDA, YSDA, CV,
  C XLP(I), UP(I)
23 FORMAT (1H , A6, 1X, 2(F11.3, 1X), 12X, 3(F11.3, 1X), 2(F11.5,1X),
```

```
     C 2(F11.3, 1X))
       GO TO 28
   24 WRITE (6, 25) IDENT(I), Y(I), X1(I), Z1(I), YCA, YDA, YSDA, CV,
     C XLP(I), UP(I)
   25 FORMAT (1H , A6, 1X, F11.3, 13X, 4(F11.3, 1X), 2(F11.5, 1X),
     C 2(F11.3, 1X))
       GO TO 28
   26 WRITE (6, 27) IDENT(I), Y(I), W1(I), X1(I), Z1(I), YCA, YDA,
     C YSDA, CV, XLP(I), UP(I)
   27 FORMAT (1H , A6, 1X, 6(F11.3, 1X), 2(F11.5, 1X), 2(F11.3, 1X))
   28 CONTINUE
       KOUNT = KOUNT + 1
       IF (KOUNT .EQ. 1) WRITE (6, 228) XLS
       IF (KOUNT .EQ. 1) GO TO 60
  228 FORMAT (1H1, 2X, 10(1H*, 1X), 11HINPUT DATA , 10(1H*, 1X), 7X,
     C 10(1H*, 1X), 30HCALCULATED DATA FOR MEAN CASE , 10(1H*, 1X) //
     C 109X, 5HLOWER, 7X, 5HUPPER / 82X, 10HSTD. ERROR, 3X, 9HCOEFF. OF,
     C 4X, 8HPREDICT., 4X,8HPREDICT./ 74X, 4HYDEV, 4X, 11HOF ESTIMATE,
     C 2X, 9HVARIATION, 5X, 5HLIMIT, 7X, 5HLIMIT / 1X, 5HLABEL, 9X, 1HY,
     C 10X, 2HX1, 10X, 2HX2, 10X, 2HX3, 9X, 7HY CALC.,3X, 10H(Y-YCALC.),
     C 1X, 10H(ADJUSTED), 3X, 9H(PERCENT), 1X, 20H(LEVEL OF SIGNIF. = ,
     C F4.2, 1H) /)
       RETURN
       END
```

```
$IBMAP W037
        REM     ROUTINE TO SET SWITCHES IN RAND W010
        REM     USAGE...CALL SETSW(N1,N2,N3,N4)
        REM                       0              1
        REM           N1      .PUF. OFF      .PUF. ON
        REM           N2      .POF. OFF      .POF. ON
        REM           N3      .KUF. OFF      .KUF. ON
        REM           N4      .KOF. OFF      .KOF. ON
        REM
        ENTRY   SETNT
        ENTRY   SETSW
        EXTERN  .PUF.,.POF.,.KUF.,.KOF.,FPTLM.
        REM
SETSW   TRA     **
        SXA     XR4,4
        LAC     SETSW,4
        CLA*    2,4             FIRST ARG
        MSP     .PUF.           ASSUME ZERO, SO TURN SWITCH OFF
        TZE     *+2             IS IT
        MSM     .PUF.           NO, SO TURN SWITCH ON
        CLA*    3,4             ETC
        MSP     .POF.
        TZE     *+2
        MSM     .POF.
        CLA*    4,4
        MSP     .KUF.
        TZE     *+2
        MSM     .KUF.
        CLA*    5,4
        MSP     .KOF.
        TZE     *+2
        MSM     .KOF.
XR4     AXT     **,4
        TRA     SETSW
SETNT   AXT     0,0
        SXA     XR4,4
        LAC     SETNT,4
        CLA*    2,4
        STO     FPTLM.
        LXA     XR4,4
        TRA*    SETNT
        END
```

```
$IBFTC QSER
C               DATE OF WRITE-UP - 3-19-65
C               DATE OF SOURCE DECK - 3-18-65
      SUBROUTINE SEARCH(KA,L,KB)
C
      DIMENSION KB(1)
      DO 1 I = 1,100
        IF (KB(I).EQ.(-34359738367)) GO TO 2
        IF(KA.EQ.KB(I)) GO TO 2
    1 CONTINUE
      I = 1
    2 L=I
      RETURN
      END
```

```
$IBMAP FPTCTL
       ENTRY   FPTCTL
       EXTERN  .POF.
FPTCTL EQU     A1
A1     PZE     **
A5     BRA     A4          FIRST TIME SWITCH
A7     NULL
       SXA     A2,4        SAVE XR4
       LAC     A1,4        -(LOC(CALL)+1)
       CLA*    2,4         ARG 1
       TZE     A8          IF ARG ZERO TRA
       STA     A3          NON-ZERO ARG
       CAL     A3          SET FPT TO GO TO
       SLW     .POF.       USER ON OVERFLOW
A2     AXT     **,4        RESTORE XR4
       TRA*    A1          RETURN
A3     TRA     **          GOES TO S.SFPT
A4     NULL
       MSP     A5          SET FIRST TIME SWITCH TO BRN
       CAL     .POF.       GET S.SFPT
       SLW     A6          SAVE IT
       TRA     A7          GO ON
A8     NULL
       CAL     A6          RESET FPT OVERFLOW LOGIC
       SLW     .POF.
       TRA     A2          RETURN
A6     ***     **
       END
```

```
$IBFTC GRTF
      SUBROUTINE GRT  (N,C,IN,IF)
      DIMENSION C(50)
      DO 100 L=1,N
      JK=0
      IF (C(L))45,46,45
 45   RT=.9*C(L)
      ASSIGN 1 TO NN
      GO TO 80
 1    X0=FPRT
      RT=1.1*C(L)
      ASSIGN 2 TO NN
      GO TO 80
 2    X1=FPRT
      RT=C(L)
      ASSIGN 3 TO NN
      GO TO 80
 3    X2=FPRT
      GO TO 50
 46   RT=-1.0
      ASSIGN 4 TO NN
      GO TO 80
 4    X0=FPRT
      RT=1.0
      ASSIGN 5 TO NN
      GO TO 80
 5    X1=FPRT
      RT=0.0
      ASSIGN 6 TO NN
      GO TO 80
 6    X2=FPRT
 50   H=-1.0
      D=-.5
 49   DD=1.0+D
      BI=(X0*D*D)-(X1*DD*DD)+(X2*(DD+D))
      DEN=BI*BI-(4.0*X2*D*DD)*(X0*D-(X1*DD)+X2)
      IF (DEN)36,36,51
 36   DEN=0.0
 51   DEN=SQRT(DEN)
 53   DN=BI+DEN
      DM=BI-DEN
      IF (ABS(DN)-ABS(DM))57,57,56
 56   DEN=DN
      GO TO 58
 57   DEN=DM
 58   IF (DEN)55,54,55
 54   DEN=1.0
 55   DI=(-2.0*X2*DD)/DEN
      H=DI*H
      RT=RT+H
      IF (ABS(H/RT)-1.0E-6)75,75,60
 60   ASSIGN 7 TO NN
      GO TO 80
 7    IF (ABS(FPRT)-ABS(X2*10.0))62,61,61
 61   DI=DI*.5
      H=H*.5
```

```
      RT=RT-H
      GO TO 80
62    X0=X1
      X1=X2
      X2=FPRT
      D=DI
      GO TO 49
75    CALL AUX (RT,FRT)
76    C(L)=RT
100   CONTINUE
      IN=JK
33    RETURN
80    JK=JK+1
      IF (100-JK)75,75,86
86    CALL AUX (RT,FRT)
      FPRT=FRT
      IF (L-1)81,89,81
81    DO 82 I=2,L
      TEM=RT-C(I-1)
      IF (ABS(TEM)-1.0E-20)85,82,82
82    FPRT=FPRT/TEM
89    IF (ABS(FRT)-1.0E-20)90,91,91
90    IF (ABS(FPRT)-1.0E-20)76,91,91
  91  IF(IF) 33,84,33
84    GO TO NN,(1,2,3,4,5,6,7)
85    RT=RT+.001
88    GO TO 80
      END
```

```
$IBFTC MATRXI
C       DATE OF WRITE-UP - 4-8-64
C       DATE OF SOURCE DECK - 4-10-65
CMATINV
C      MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF LINEAR EQUATIONS
C
       SUBROUTINE MATINV (A,N,B,M,DETERM,PIVOT,INDEX,IPIVOT)
C
       DIMENSION IPIVOT(50),A(50,50),B(50,1),INDEX(50,2),PIVOT(50)
       EQUIVALENCE (IROW,JROW), (ICOLUM,JCOLUM), (AMAX, T, SWAP)
C
C      INITIALIZATION
C
    10 DETERM=1.0
    15 DO 20 J=1,N
    20 IPIVOT(J)=0
    30 DO 550 I=1,N
C
C      SEARCH FOR PIVOT ELEMENT
C
    40 AMAX=0.0
    45 DO 105 J=1,N
    50 IF (IPIVOT(J)-1) 60, 105, 60
    60 DO 100 K=1,N
    70 IF (IPIVOT(K)-1) 80, 100, 720
    80 IF (ABS(AMAX)-ABS(A(J,K))) 85, 100, 100
    85 IROW=J
    90 ICOLUM=K
    95 AMAX=A(J,K)
   100 CONTINUE
   105 CONTINUE
          IF(AMAX) 110,106,110
   106 IPIVOT(1) = -1
   107 PRINT 108
   108 FORMAT(17HODETERMINANT = 0. )
   109    GO TO 740
   110 IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1
C
C      INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL
C
   130 IF (IROW-ICOLUM) 140, 260, 140
   140 DETERM=-DETERM
   150 DO 200 L=1,N
   160 SWAP=A(IROW,L)
   170 A(IROW,L)=A(ICOLUM,L)
   200 A(ICOLUM,L)=SWAP
   205 IF(M) 260, 260, 210
   210 DO 250 L=1, M
   220 SWAP=B(IROW,L)
   230 B(IROW,L)=B(ICOLUM,L)
   250 B(ICOLUM,L)=SWAP
   260 INDEX(I,1)=IROW
   270 INDEX(I,2)=ICOLUM
   310 PIVOT(I)=A(ICOLUM,ICOLUM)
   320 DETERM=DETERM*PIVOT(I)
C
```

```
C      DIVIDE PIVOT ROW BY PIVOT ELEMENT
C
  330 A(ICOLUM,ICOLUM)=1.0
  340 DO 350 L=1,N
  350 A(ICOLUM,L)=A(ICOLUM,L)/PIVOT(I)
  355 IF(M) 380, 380, 360
  360 DO 370 L=1,M
  370 B(ICOLUM,L)=B(ICOLUM,L)/PIVOT(I)
C
C      REDUCE NON-PIVOT ROWS
C
  380 DO 550 L1=1,N
  390 IF(L1-ICOLUM) 400, 550, 400
  400 T=A(L1,ICOLUM)
  420 A(L1,ICOLUM)=0.0
  430 DO 450 L=1,N
  450 A(L1,L)=A(L1,L)-A(ICOLUM,L)*T
  455 IF(M) 550, 550, 460
  460 DO 500 L=1,M
  500 B(L1,L)=B(L1,L)-B(ICOLUM,L)*T
  550 CONTINUE
C
C      INTERCHANGE COLUMNS
C
  600 DO 710 I=1,N
  610 L=N+1-I
  620 IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630
  630 JROW=INDEX(L,1)
  640 JCOLUM=INDEX(L,2)
  650 DO 705 K=1,N
  660 SWAP=A(K,JROW)
  670 A(K,JROW)=A(K,JCOLUM)
  700 A(K,JCOLUM)=SWAP
  705 CONTINUE
  710 CONTINUE
          GO TO 740
  720 PRINT 725
  725 FORMAT(23HOPROBABLE MACHINE ERROR)
  730 CALL EXIT
  740 RETURN
  750 END
```

```
$IBFTC EV
      SUBROUTINE EIGEN(A,VALU,N,M,B,DIAG,SUPERD,Q,VALL,S,D,C,U,IND,
     C NOEIG)
C     EIGENVALUES AND EIGENVECTORS OF A REAL SYMMETRIC MATRIX
      DIMENSION  A(50,50), B(50,50), VALU(50), DIAG(50), SUPERD(49),
     C Q(49), VALL(50), S(49), C(49), D(50), IND(50), U(50)
      DIMENSION P(53), R(51), RATIO(52), INI(52)
      EQUIVALENCE (II, MATCH), (TAU, BETA), (P,PRODS), (T, SMALLD),
     C (ANORM, ANORM2)
C     CALCULATE NORM OF MATRIX
C
    2 NVK = N
      EVNORM = 1.0E-8
      SIMAVG = 1.0E-7
      EVSEP = 1.0E-4
      NOEIG = 0
    3 ANORM2=0.0
    4 DO 6 I=1,N
    5 DO 6 J=1,N
    6 ANORM2=ANORM2+A(I,J)**2
    7 ANORM=SQRT(ANORM2)
C
C     GENERATE IDENTITY MATRIX
C
    9 IF (M) 10, 45, 10
   10 DO 40 I=1,N
   12 DO 40 J=1,N
   20 IF(I-J) 35, 25, 35
   25 B(I,J)=1.0
   30 GO TO 40
   35 B(I,J)=0.0
   40 CONTINUE
C
C     PERFORM ROTATIONS TO REDUCE MATRIX TO JACOBI FORM
C
   45 IEXIT=1
   50 NN=N-2
   52 IF (NN) 890, 170, 55
   55 DO 160 I=1,NN
   60 II=I+2
   65 DO 160 J=II,N
   70 T1=A(I,I+1)
   75 T2=A(I,J)
   80 GO TO 900
   90 DO 105 K=I,N
   95 T2=COS*A(K,I+1)+SIN*A(K,J)
  100 A(K,J)=COS*A(K,J)-SIN*A(K,I+1)
  105 A(K,I+1)=T2
  110 DO 125 K=I,N
  115 T2=COS*A(I+1,K)+SIN*A(J,K)
  120 A(J,K)=COS*A(J,K)-SIN*A(I+1,K)
  125 A(I+1,K)=T2
  128 IF (M) 130, 160, 130
  130 DO 150 K=1,N
  135 T2=COS*B(K,I+1)+SIN*B(K,J)
  140 B(K,J)=COS*B(K,J)-SIN*B(K,I+1)
```

```
  150 B(K,I+1)=T2
  160 CONTINUE
C
C     MOVE JACOBI FORM ELEMENTS AND INITIALIZE EIGENVALUE BOUNDS
C
  170 DO 200 I=1,N
  180 DIAG(I)=A(I,I)
  190 VALU(I)=ANORM
  200 VALL(I)=-ANORM
      DO 230 I=2,N
  220 SUPERD(I-1)=A(I-1,I)
  230 Q(I-1)=(SUPERD(I-1))**2
C
C     DETERMINE SIGNS OF PRINCIPAL MINORS
C
  235 TAU=0.0
  240 I=1
      NFI = 0
  260 MATCH=0
  270 T2=0.0
  275 T1=1.0
  277 DO 450 J=1,N
  280 P(1) = DIAG(J) - TAU
  290 IF(T2) 300, 330, 300
  300 IF(T1) 310, 370, 310
  310 T = P(1) * T1 - Q(J-1) * T2
      IF (T.NE.0.0) GO TO 410
      PT1 = ABS(P(1)*T1)
      QT2 = ABS(Q(J-1)*T2)
      IF (AMAX1(PT1, QT2).GT.1.0E-30) GO TO 410
      T1 = T1 * 1.0E20
      T2 = T2 * 1.0E20
      GO TO 310
  330 IF(T1) 335, 350, 350
  335 T1=-1.0
  340 T = -P(1)
  345 GO TO 410
  350 T1=1.0
  355 T = P(1)
  360 GO TO 410
  370 IF(Q(J-1)) 380, 350, 380
  380 IF(T2) 400, 390, 390
  390 T=-1.0
  395 GO TO 410
  400 T=1.0
C
C     COUNT AGREEMENTS IN SIGN
C
  410 IF(T1) 425, 420, 420
  420 IF(T) 440, 430, 430
  425 IF(T) 430, 440, 440
  430 MATCH=MATCH+1
  440 T2=T1
  450 T1=T
C
C     ESTABLISH TIGHTER BOUNDS ON EIGENVALUES
```

```
C
  460 DO 530 K=1,N
  465 IF (K-MATCH) 470, 470, 520
  470 IF(TAU-VALL(K)) 530, 530, 480
  480 VALL(K)=TAU
  490 GO TO 530
  520 IF(TAU-VALU(K)) 525, 530, 530
  525 VALU(K)=TAU
  530 CONTINUE
  540 IF(VALU(I)-VALL(I)-5.0E-8) 570, 570, 550
  550 IF(VALU(I)) 560, 580, 560
  560 IF(ABS(VALL(I)/VALU(I)-1.0)-5.0E-8) 570, 570, 580
C                   COMPARE EIGENVALUE AGAINST MINIMUM THRESHOLD
  570 NFI = I
  575 NFI = I
      I = I+1
      IF(I-N) 540, 540, 590
  580 TAU=(VALL(I)+VALU(I))/2.0
  585 GO TO 260
C
C     JACOBI EIGENVECTORS BY ROTATIONAL TRIANGULARIZATION
C
  590 IF (M) 593, 890, 593
  593 IEXIT=2
  595 DO 610 I=1,N
  600 DO 610 J=1,N
  610 A(I,J)=0.0
  615 DO 850 I = 1, NFI
  620 IF (I-1) 625, 625, 621
  621 IF ( VALU(I-1) - VALU(I) -EVSEP) 730, 730, 622
  622 IF (VALU(I-1)) 623, 625, 623
  623 IF (ABS(VALU(I)/VALU(I-1) - 1.0) -EVSEP) 730, 730, 625
  625 COS=1.0
  628 SIN=0.0
  630 DO 700 J=1,N
  635 IF (J-1) 680, 680, 900
  650 S(J-1)=SIN
  660 C(J-1)=COS
  670 D(J-1)=T1*COS+T2*SIN
  680 T1=(DIAG(J)-VALU(I))*COS-BETA*SIN
  690 T2=SUPERD(J)
  700 BETA=SUPERD(J)*COS
  710 D(N)=T1
  720 DO 725 J=1,N
  725 IND(J)=0
  730 SMALLD=ANORM
  735 DO 780 J=1,N
  740 IF (IND(J)-1) 750, 780, 780
  750 IF (ABS(SMALLD)-ABS(D(J)))780, 780, 760
  760 SMALLD=D(J)
  770 NN=J
  780 CONTINUE
  790 IND(NN)=1
  800 PRODS=1.0
  805 IF (NN-1) 810, 850, 810
  810 DO 840 K=2,NN
```

```
  820 II=NN+1-K
  830 A(II+1,I)=C(II)*PRODS
  840 PRODS=-PRODS*S(II)
  850 A(1,I)=PRODS
 1000 J = 1
C                    COMPUTE LENGTH OF FIRST EIGENVECTOR
      TEMP = 0.
      DO 1005 K = 1, NVK
 1005 TEMP = TEMP + A(K,1) * A(K,1)
      IF (ABS(TEMP - 1.0) - EVNORM) 1025, 1025, 1010
C                    NORMALIZE FIRST EIGENVECTOR TO UNIT LENGTH
 1010 TEMP = SQRT (TEMP)
      DO 1020 K = 1, NVK
 1020 A(K,1) = A(K,1) / TEMP
 1025 L = 0
      SQSUM = 2.0
      SUPN = 1.0
C                    COMPUTE (JACOBI - VALU(J) * I) * A (I,J)
 1030 U(1) = (DIAG (1) - VALU (J)) * A(1,J) + SUPERD(1) * A(2,J)
      DIFSUP= ABS(U(1))
      DIFSQ = DIFSUP ** 2
      NN = N-1
      DO 1040 I = 2, NN
      U(I) = SUPERD(I-1) * A(I-1,J) +(DIAG(I) - VALU(J)) * A(I,J)
     1       + SUPERD(I) * A(I+1, J)
      TEMP = ABS(U(I))
      DIFSQ = DIFSQ + TEMP * TEMP
 1040 DIFSUP=AMAX1 (DIFSUP, TEMP)
      U(N) = SUPERD(NN) * A(NN,J) + (DIAG(N) - VALU(J)) * A(N,J)
      TEMP = ABS (U(N))
      DIFSQ = DIFSQ + TEMP * TEMP
      DIFSUP=AMAX1 (DIFSUP, TEMP)
      DIFSQ = SQRT (DIFSQ)
      IF (DIFSQ.GT.SIMAVG) GO TO 1045
      IF (IEXIT.EQ.1.OR.J.EQ.1) GO TO 1400
      IF (VALU(J-1)-VALU(J).GT.EVSEP) GO TO 1400
      J = J-1
      GO TO 1430
 1045 IF (IEXIT.NE.1) GO TO 1055
      IF (DIFSQ.GE.SMALLD) GO TO 1400
 1050 SMALLD = DIFSQ
 1055 IF (SQSUM) 1070, 1060, 1070
 1060 EV1 = 0.0
      DO 1061 I = 1, NVK
      U(I) = U(I) + VALU(J) * A(I,J)
 1061 EV1 = EV1 + A(I,J) * U(I)
      DIFSQ1 = 0.0
      DO 1062 I = 1, NVK
 1062 DIFSQ1 = DIFSQ1 + (U(I) - EV1 * A(I,J)) ** 2
      DIFSQ1 = SQRT (DIFSQ1)
      IF (EV1  .GE.VALU(J+1)) GO TO 1065
      EV1    = 0.5 * (VALU(J) + VALU(J+1))
 1065 IF (DIFSQ1.GE.DIFSQ*0.9) GO TO 1070
      L = 0
      SQSUM = 2.0
      SUPN = 1.0
```

```
      VALU (J) = EV1
 1070 IF (L) 1155, 1100, 1155
C                  PERFORM FORWARD SOLUTION FOR JACOBI MATRIX - VALU(J)*I
 1100 PTEMP = DIAG(1) - VALU(J)
      QTEMP = SUPERD(1)
      DO 1150 K = 1, NN
      IF (ABS(PTEMP) - ABS(SUPERD(K))) 1110, 1110, 1130
 1110 INI(K) = 1
      P(K) = SUPERD(K)
      Q(K) = DIAG(K+1) - VALU(J)
      RATIO(K)=PTEMP/SUPERD(K)
      PTEMP = QTEMP -RATIO(K)*Q(K)
      IF (K-NN) 1120, 1150, 1150
 1120 R(K) = SUPERD(K+1)
      QTEMP =-RATIO(K)*SUPERD(K+1)
      GO TO 1150
 1130 INI(K) = 0
      P(K) = PTEMP
      Q(K) = QTEMP
      RATIO(K)=SUPERD(K)/PTEMP
      PTEMP = DIAG(K+1) - VALU(J)-RATIO(K)*QTEMP
      IF (K - NN) 1140, 1150, 1150
 1140 R(K) = 0
      QTEMP = SUPERD(K+1)
 1150 CONTINUE
      P(N) = PTEMP
 1155 DO 1160 I = 1, N
 1160 U(I) = A(I,J)
C.........OPERATE ON LTH APPROX WITH SEQ OF LIN OPS FROM FWD SOL
 1200 DO 1230 I = 1, NN
      IF (INI(I)) 1210, 1220, 1210
 1210 TEMP = U(I)
      U(I) = U(I+1)
      U(I+1) = TEMP - RATIO(I) * U(I+1)
      GO TO 1230
 1220 U(I+1) = U(I+1) - RATIO(I) * U(I)
 1230 CONTINUE
C                  PERFORM BACK SUBSTITUTION TO SOLVE (J-LAMBDA*I)*X = U
      U(N) = U(N) / P(N)
      U(N-1) = (U(N-1) - Q(N-1) * U(N)) / P(N-1)
      TEMP = U(N) * U(N) +  U(N-1) * U(N-1)
      DO 1240 K = 2, NN
      NK = N-K
      U(NK) = (U(NK) - Q(NK) * U(NK+1) - R(NK) * U(NK+2)) / P(NK)
 1240 TEMP = TEMP + U(NK) * U(NK)
      TEMP = SQRT(TEMP)
C                  NORMALIZE SOLUTION
      DO 1260 K = 1, NVK
 1260 U(K) = U(K) / TEMP
C                  COMPARE NORMALIZED SOLUTION WITH PREVIOUS ESTIMATE
      DIFSQ = 0.
      DIFSUP = 0.
      DO 1270 I = 1, N
      TEMP = ABS (U(I)) - ABS (A(I,J))
      DIFSQ = DIFSQ + TEMP ** 2
      TEMP = ABS(TEMP)
```

```
 1270 DIFSUP = AMAX1 (DIFSUP, TEMP)
      DIFSQ = SQRT(DIFSQ)
      IF (L.LT.5) GO TO 1300
      IF (DIFSQ - SQSUM) 1280, 1290, 1290
 1280 IF (DIFSUP - SUPN) 1300, 1300, 1290
 1290 NOEIG = 1
      WRITE (6, 9040) J
      WRITE (6, 9050) L, DIFSQ, DIFSUP
      IF (J-1) 1420, 1420, 1295
 1295 NFI = J-1
      GO TO 855
C                     REPLACE LTH APPROXIMATION BY (L+1)TH FOR JTH EIGENVEC.
 1300 DO 1310 I = 1, N
 1310 A(I,J) = U(I)
      IEXIT = 2
      IF (DIFSQ.LE.SIMAVG) GO TO 1330
      IF (DIFSQ.GT.SQSUM * .9) GO TO 1330
      SQSUM = DIFSQ
      SUPN = DIFSUP
 1325 L = L+1
      GO TO 1030
 1330 SQSUM = 0.
      GO TO 1325
C                     GRAM-SCHMIDT ORTHOGONALIZATION
 1400 IF (J - NFI) 1410, 855, 855
 1410 SMALLD = 2.0
      GO TO 1430
 1420 NFI = J
      GO TO 855
C                     COMPUTE PROJECTIONS OF (J+1)TH VECTOR ON FIRST J
 1430 DO 1440 K = 1, J
      C(K) = 0.
      JK = J+1-K
      DO 1440 I = 1, NVK
 1440 C(K) = C(K) + A(I, J+1) * A(I, JK)
C                     REPLACE (J+1)TH EIGENVECTOR BY ORTHOGONALIZED VECTOR
 1460 DO 1470 K = 1, J
      JK = J+1-K
      DO 1470 I = 1, NVK
 1470 A(I, J+1) =  A(I, J+1) - C(K) * A(I, JK)
C                     NORMALIZE THE ORTHOGONALIZED (J+1)TH EIGENVECTOR
      TEMP = 0.
      DO 1480 I = 1, NVK
 1480 TEMP = TEMP + A(I,J+1)** 2
      IF (TEMP.NE.0.0) GO TO 1490
      GO TO 855
 1490 J= J+1
      TEMP = SQRT (TEMP)
      DO 1500 I = 1, NVK
 1500 A(I,J) = A(I,J)/TEMP
 1510 IEXIT = 1
      GO TO 1025
C
C     FORM MATRIX PRODUCT OF ROTATION MATRIX WITH JACOBI VECTOR MATRIX
C
  855 DO 885 J = 1, NFI
```

```
  860 DO 865 K=1,N
  865 U(K)=A(K,J)
  870 DO 885 I=1,N
  875 A(I,J)=0.0
  880 DO 885 K=1,N
  885 A(I,J)=B(I,K)*U(K)+A(I,J)
  890 RETURN
C
C     CALCULATE SINE AND COSINE OF ANGLE OF ROTATION
C
  900 IF (T2) 910, 940, 910
  910 T=SQRT(T1**2+T2**2)
  920 COS=T1/T
  925 SIN=T2/T
  930 GO TO (90,650), IEXIT
  940 GO TO (160,910), IEXIT
 9040 FORMAT (71HOFOR THE DIFFERENCE BETWEEN APPROXIMATIONS L AND L+1 TO
     1 EIGENVECTOR NO., I6)
 9050 FORMAT (9HO.... L =, I6, 5X, 19HLENGTH OR L2 NORM =, F12.8, 5X,
     133HMAXIMUM MAGNITUDE OF COMPONENTS =, F12.8)
  950 END
```

REFERENCES

1. Bennett, Carl A., and Norman L. Franklin, Statistical Analysis in Chemistry and the Chemical Industry, John Wiley & Sons, Inc., New York, 1954.

2. Gabler, R. T., and Albert Madansky, On General Least Squares Estimation of Solar System Constants, The RAND Corporation, RM-4019-NASA, February 1964.

3. Hartley, O. H., "The Modified Gauss-Newton Method for the Fitting of Non-Linear Regression Functions by Least Squares," Technometrics, Vol. 3, No. 2, 1961.

4. Hohn, Franz E., Elementary Matrix Algebra, The Macmillan Company, New York, 1958.

5. Lindgren, B. W., Statistical Theory, The Macmillan Company, New York, 1962.

6. Wallace, David L., "Bounds on Normal Approximations to Student's and the Chi-square Distributions," Ann. Math. Stat., Vol. 30, No. 4, December 1959, pp. 1121-1130.