THE EFFECT OF BIAS ON LEAST SQUARES ESTIMATORS OF THE SLOPE PARAMETER

By

JULIA KAY SAWYER

Bachelor of Science
Northwestern Oklahoma State University
Alva, Oklahoma
1991

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1993

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
May, 1996

THE EFFECT OF BIAS ON LEAST

SQUARES ESTIMATORS OF

THE SLOPE PARAMETER

Thesis Approved:

Barry Kurt Moser
Thesis Adviser

P. Larry Claypool

Mark E Payton

W M Warde

D B Aichele

Thomas C. Collins
Dean of the Graduate College

ACKNOWLEDGEMENTS

The educational process is a journey and no one walks alone on this particular pathway. I would like to sincerely thank all my fellow travellers helping me to my destination. I would like to express my deepest appreciation and respect for my adviser, Dr. Barry Kurt Moser. He has been my guide throughout my graduate travels, offering support and inspiration at every turn. I thank him for his invitation to learn.

I would like to thank the members of my committee - Dr. Larry Claypool, Dr. Bill Warde, Dr. Mark Payton and Dr. Doug Aichele - for their advice, encouragement and friendship. I am grateful to Dr. Mindy McCann for her computer assistance.

Special thanks go to Scott Richter and Carri Duvall for being proofreaders, surrogate siblings, and very good friends. I thank Fran Mihura and Becki Moore for taking care of me when I forgot to take care of myself. My love and appreciation also go to Kelly Ridgway, just because.

My eternal gratitude goes to my son, Luke Faires, for his love and his patience. Most importantly, I would like to thank my husband, Keith, for everything.

Thank you for the shared laughter and tears. I love you all.

TABLE OF CONTENTS

Chapter Page

APPENDICES (continued)

LIST OF TABLES

LIST OF FIGURES

NOMEMCLATURE

WLSE	weighted least squares estimator of β_1
OLSE	ordinary least squares estimator of β_1
OLSBE	ordinary least squares between estimator of β_1
OLSWE	ordinary least squares within estimator of β_1
γ	ratio of within sums of squares to total sums of squares for $\mathbf{X}_1$
ρ	ratio of variance components
λ	ratio of squared bias of ordinary least squares estimator of β_1 to the variance of the ordinary least squares within estimator of β_1
RE	relative efficiency

CHAPTER I

INTRODUCTION

In the estimation of slope parameters in a regression context, a major consideration is unbiasedness of the estimator. The ordinary and weighted least squares estimators are often used when the form of the linear model is assumed known. The purpose of this study is to explore options for determining the best estimator for the slope of the regression line when the researcher suspects the existence of an unknown bias term in the model.

The layout of the repeated measures design to be considered is illustrated in Figure 1 on the following page. The independent variable, X_{ijk}, is measured along with the dependent variable, Y_{ijk}, over time where i denotes the level for the classification, j designates the replication number within each level of i and k denotes the level for time. Observations are considered replications if they are within the same classification and if they have the same bias.

Consider the centered model

$$Y_{ijk} = \beta_0 + \beta_1(x_{ijk}-\bar{x}_{...}) + s_i + \varepsilon_{ijk}, \qquad (1)$$

where β_0 is the unknown intercept, β_1 is the unknown slope parameter, $\bar{x}_{...}$ is the overall mean of the x_{ijk}'s, s_i is the bias in the i^{th} classification level, and ε_{ijk} is the random error such that $E(\varepsilon_{ijk}) = 0$ for all i = 1, 2, ..., a, j = 1, Classification

TIME	Rep		1		...		a	
		1	...	r	...	1	...	r
1		X_{111}, Y_{111}	...	$X_{1r_1 1}$, $Y_{1r_1 1}$	...	X_{a11}, Y_{a11}	...	$X_{ar_a 1}$, Y_{ar1}
2		X_{112}, Y_{112}	...	$X_{1r_1 2}$, $Y_{1r_1 2}$	...	X_{a12}, Y_{a12}	...	$X_{ar_a 2}$, Y_{ar2}
.		.	...	.	...	.	...	.
.		.	...	.	...	.	...	.
.		.	...	.	...	.	...	.
b		$X_{11b_{11}}$, $Y_{11b_{11}}$	...	$X_{1r_1 b_{1r}}$, $Y_{1r_1 b_{1r}}$	...	$X_{a1b_{a1}}$, $Y_{a1b_{a1}}$	...	$X_{ar_a b_{ar}}$, $Y_{ar_a b_{ar}}$

Figure 1. Layout of the experiment

2, ... , r_i, and k = 1, 2, ..., b_{ij}. Then $\sum_{i=1}^{a}\sum_{j=1}^{r_i} b_{ij} = n$, where n is the total number of observations. In matrix form, the model is

$$\mathbf{Y} = \mathbf{1}_n\beta_0 + \mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)} + \varepsilon, \qquad (2)$$

where $\mathbf{X}_1$ is an *nx1* vector such that $\mathbf{X}_1 = (x_{111}-\bar{x}_{...}, \ldots, x_{11b_{r_1}}-\bar{x}_{...}, \ldots, x_{ar_a1}-\bar{x}_{...}, \ldots, x_{ar_ab_{r_a}}-\bar{x}_{...})'$, $\mathbf{X}_2$ is an *nx(p-2)* matrix which is centered so that $\mathbf{1}_n'\mathbf{X}_2 = 0$, $\beta^{(2)}$ is a *(p-2)x1* vector such that $\beta^{(2)} = (\beta_2, \beta_3, \ldots, \beta_{p-1})'$, and ε is the *nx1* vector of random errors. The $\mathbf{X}_2\beta^{(2)}$ term represents the bias in the model. The bias within each classification level, s_i, is constant over time and replication.

Let $\mathbf{X} = (\mathbf{1}_n|\mathbf{X}_1)$. If the assumption $s_i = 0$ for all i is made, i.e. there is no significant bias, the ordinary least squares estimator (OLSE) of the parameters $(\beta_0 \ \ \beta_1)'$,

$$\begin{bmatrix}\hat{\beta}_0 \\ \hat{\beta}_1\end{bmatrix} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y},$$

is the best linear unbiased estimator when the simple covariance structure $\Sigma = \sigma_\varepsilon^2\mathbf{I}_n$ is assumed.

The slope estimator, $\hat{\beta}_1$, remains unbiased when the bias is constant across levels of classification, $s_i = c$ for all i. The estimator for the intercept, $\hat{\beta}_0$, is affected when bias is constant but non-zero, as the least squares line is moved up or down by the value of the constant.

When a more complicated covariance structure, say $\Sigma = \mathbf{V}$ is utilized, the weighted least squares estimator (WLSE) of

$(\beta_0 \quad \beta_1)'$,

$$\begin{bmatrix} \hat{\beta}_0^* \\ \hat{\beta}_1^* \end{bmatrix} = (\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{V}^{-1}\mathbf{Y},$$

is the best linear unbiased estimator for the parameters when $s_i = 0$, or if only β_1 is considered, when $s_i = c$.

If a situation arises in which the bias changes across levels of classification, the least squares estimators of β_1 become biased. Hence, it is of interest to find a best estimator of β_1 when bias is present. To this end, the WLSE, and therefore the OLSE, of β_1 will be shown to be a linear combination of a within classifications estimator of β_1 and a between classifications estimator of β_1. The behavior of the OLSE, WLSE, the between estimator and the within estimator will be compared when bias is present in order to determine a best estimator for given situations.

A search of the literature for work in this area produced little information, due to the narrowness of the problem under investigation. Draper and Smith discuss the effect of bias on the least squares analysis and give the form of the analysis of variance (ANOVA) table. General resources in the areas of regression, linear models, and estimation of variance components were utilized and are listed in the bibliography.

The objective of Chapter II is to derive the form of the weighted least squares, between classifications and within classifications estimators of β_1 for model (1).

Special cases will be discussed, including the consideration of the ordinary least squares estimator as a special case of the weighted least squares estimator in which $\mathbf{V} = \sigma_{\varepsilon}^{2}\mathbf{I}_{n}$.

The objective of Chapter III is to discuss the quality of the OLS, WLS, between classifications and within classifications estimators of β_1. The effect of bias in the model on the biasedness of the estimators will be investigated, and best linear unbiasedness will be discussed in relation to the covariance structure.

The derivation of variance estimators is the objective of Chapter IV. The objective of Chapter V is to compare the mean square errors of the estimators of β_1. The relative efficiencies are calculated and used to construct tables for determining the best estimator for sets of fixed parameter values. In addition, a method for estimating the bias parameter is suggested.

In Chapter VI results of simulations will be discussed, including a determination of how often the correct estimator is selected using the tables. It is assumed that an estimator for λ is available. Concluding remarks and suggestions for future research are offered in Chapter VII.

CHAPTER II

DERIVATION OF SLOPE ESTIMATORS

The between classifications estimator and within classifications estimator of β_1 are derived by showing that the weighted least squares estimator is a linear combination of a between component and a within component. Assume the data structure is as described in Chapter I. Also assume that the bias is unknown.

Let $\mathbf{X} = (\mathbf{1}_n | \mathbf{X}_1)$ where $\mathbf{X}_1 = (x_{111} - \bar{x}_{...}, \ldots, x_{11b_{r_1}} - \bar{x}_{...}, \ldots, x_{ar_a1} - \bar{x}_{...}, \ldots, x_{ar_ab_{r_a}} - \bar{x}_{...})'$. Also, let $\mathbf{Y} = (y_{111}, \ldots, y_{11b_{11}}, \ldots, y_{ar_a1}, \ldots, y_{ar_ab_{ar_a}})'$.

Direct sum notation will be utilized to designate certain matrices. Let $\mathbf{A}_{ij}$ be matrices of dimensions $a_i x b_j$, i = 1,...u and j = 1,...,v. Then the nxn block diagonal matrix $\mathbf{A}$ is as follows:

$$\mathbf{A}=\begin{bmatrix}
\mathbf{A}_{11} & \mathbf{0} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{0} \\
\mathbf{0} & \cdot & \mathbf{0} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \mathbf{0} & \mathbf{A}_{1b_1} & \cdot & \mathbf{0} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \mathbf{0} & \cdot & \mathbf{0} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \mathbf{0} & \mathbf{A}_{21} & \mathbf{0} & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \mathbf{0} & \cdot & \mathbf{0} & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{0} & \mathbf{A}_{2b_2} & \mathbf{0} & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{0} & \cdot & \mathbf{0} & \mathbf{0} & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{0} & \mathbf{A}_{a_u1} & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{0} & \cdot & \mathbf{0} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{0} & \mathbf{A}_{a_ub_v}
\end{bmatrix}$$

$= \bigoplus_{i=1}^{u} \bigoplus_{j=1}^{v} (\mathbf{A}_{ij})$. Then define to be $\mathbf{R}_1 = \bigoplus_{i=1}^{a} \bigoplus_{j=1}^{r_i} (\mathbf{I}_{r_i} \otimes b_{ij}^{-1} \mathbf{J}_{b_{ij}})$ and $\mathbf{R}_2 = \bigoplus_{i=1}^{a} \bigoplus_{j=1}^{r_i} [\mathbf{I}_{r_i} \otimes (\mathbf{I}_{b_{ij}} - b_{ij}^{-1} \mathbf{J}_{b_{ij}})]$ for i = 1, ..., a and j = 1, ..., r_i, which will be utilized in the derivation of the slope estimators.

The weighted least squares estimator of $(\beta_0, \beta_1)'$ is given by

$$\begin{pmatrix} \hat{\beta}_0^* \\ \hat{\beta}_1^* \end{pmatrix} = (\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{V}^{-1}\mathbf{Y}$$

$$= \left[\begin{pmatrix} \mathbf{1}_n' \\ \mathbf{X}_1' \end{pmatrix}\mathbf{V}^{-1}(\mathbf{1}_n|\mathbf{X}_1)\right]^{-1}\begin{pmatrix} \mathbf{1}_n' \\ \mathbf{X}_1' \end{pmatrix}\mathbf{V}^{-1}\mathbf{Y}$$

$$= \begin{pmatrix} \mathbf{1}_n'\mathbf{V}^{-1}\mathbf{1}_n & \mathbf{1}_n'\mathbf{V}^{-1}\mathbf{X}_1 \\ \mathbf{X}_1'\mathbf{V}^{-1}\mathbf{1}_n & \mathbf{X}_1'\mathbf{V}^{-1}\mathbf{X}_1 \end{pmatrix}^{-1}\begin{pmatrix} \mathbf{1}_n' \\ \mathbf{X}_1' \end{pmatrix}\mathbf{V}^{-1}\mathbf{Y}$$

$$= \begin{pmatrix} (\mathbf{1}_n'\mathbf{V}^{-1}\mathbf{1}_n)^{-1} & 0 \\ 0 & (\mathbf{X}_1'\mathbf{V}^{-1}\mathbf{X}_1)^{-1} \end{pmatrix}\begin{pmatrix} \mathbf{1}_n' \\ \mathbf{X}_1' \end{pmatrix}\mathbf{V}^{-1}\mathbf{Y}$$

$$= \begin{pmatrix} (\mathbf{1}_n'\mathbf{V}^{-1}\mathbf{1}_n)^{-1}\mathbf{1}_n'\mathbf{V}^{-1}\mathbf{Y} \\ (\mathbf{X}_1'\mathbf{V}^{-1}\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{V}^{-1}\mathbf{Y} \end{pmatrix}.$$

Consider only $\hat{\beta}_1^*$, the estimator for the slope, β_1. This can be written as a linear combination of two components. Assuming $\mathbf{V} = \sigma_R^2[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(\mathbf{I}_{r_i}\otimes\mathbf{J}_{b_{ij}})] + \sigma_\varepsilon^2\mathbf{I}_n$,

$$\mathbf{V}^{-1} = \bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1}(\mathbf{I}_{r_i}\otimes b_{ij}^{-1}\mathbf{J}_{b_{ij}}) + \sigma_\varepsilon^{-2}\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_{b_{ij}}-b_{ij}^{-1}\mathbf{J}_{b_{ij}})].$$

Also, since $\mathbf{V}^{-1}$ is positive definite, it may be expressed as $\mathbf{PP}'$ where $\mathbf{P}$ is an *nxn* matrix

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}^*_{11} & \mathbf{0} & . & . & \mathbf{0} \\ \mathbf{0} & \mathbf{P}^*_{12} & \mathbf{0} & . & . \\ . & & . & & . \\ . & & & . & \\ \mathbf{0} & . & . & \mathbf{0} & \mathbf{P}^*_{ar_a} \end{bmatrix} = \bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}\mathbf{P}^*_{ij}$$

where $\mathbf{P}^*_{ij}$ is a $b_{ij}\mathrm{x}b_{ij}$ matrix of the form

$\mathbf{P}^*_{ij} = [(b_{ij}\sigma^2_R+\sigma^2_\varepsilon)^{-1/2}\mathbf{p}_{1ij}|\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}})]$ with $\mathbf{p}_{1ij} = b_{ij}^{-1/2}\mathbf{1}_{b_{ij}}$

and $\mathbf{P}_{b_{ij}}'$ is a $(b_{ij}-1)\mathrm{x}b_{ij}$ dimensional lower portion of a Helmert matrix (Moser, p. 6). Then

$$\mathbf{P}^*_{ij}\mathbf{P}^*_{ij}{}' = ((b_{ij}\sigma_R{}^2+\sigma^2_\varepsilon)^{-1/2}\mathbf{p}_{1ij}|\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}) \begin{bmatrix} (b_{ij}\sigma^2_R+\sigma^2_\varepsilon)^{-1/2}\mathbf{p}_{1ij}' \\ \sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}' \end{bmatrix}$$

$$= (b_{ij}\sigma^2_R+\sigma^2_\varepsilon)^{-1/2}\mathbf{p}_{1ij}(b_{ij}\sigma^2_R+\sigma^2_\varepsilon)^{-1/2}\mathbf{p}_{1ij}' + \sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}'$$

$$= (b_{ij}\sigma^2_R+\sigma^2_\varepsilon)^{-1}b_{ij}^{-1/2}\mathbf{1}_{b_{ij}}b_{ij}^{-1/2}\mathbf{1}_{b_{ij}}' + \sigma_\varepsilon^{-2}(\mathbf{I}_{b_{ij}} - b_{ij}^{-1}\mathbf{J}_{b_{ij}})$$

$$= (b_{ij}\sigma^2_R+\sigma^2_\varepsilon)^{-1}b_{ij}^{-1}\mathbf{J}_{b_{ij}} + \sigma_\varepsilon^{-2}(\mathbf{I}_{b_{ij}} - b_{ij}^{-1}\mathbf{J}_{b_{ij}}) \text{ and}$$

$$\mathbf{PP}' = \begin{bmatrix} \mathbf{P}^*_{11}\mathbf{P}^*_{11}{}' & \mathbf{0} & . & . & \mathbf{0} \\ \mathbf{0} & \mathbf{P}^*_{12}\mathbf{P}^*_{12}{}' & \mathbf{0} & . & . \\ . & & . & & . \\ . & & & . & \\ \mathbf{0} & . & . & \mathbf{0} & \mathbf{P}^*_{ar_a}\mathbf{P}^*_{ar_a}{}' \end{bmatrix}$$

$$= \bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}\mathbf{P}^*_{ij}\mathbf{P}^*_{ij}{}'$$

$$= \bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma^2_R+\sigma^2_\varepsilon)^{-1}(\mathbf{I}_{r_i}\otimes b_{ij}^{-1}\mathbf{J}_{b_{ij}}) + \sigma_\varepsilon^{-2}\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_{b_{ij}}-b_{ij}^{-1}\mathbf{J}_{b_{ij}})]$$

$$= [\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma^2_R+\sigma^2_\varepsilon)^{-1}]\mathbf{R}_1 + \sigma_\varepsilon^{-2}\mathbf{R}_2.$$

Let $\mathbf{R}^*_1 = [\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma^2_R+\sigma^2_\varepsilon)^{-1}]\mathbf{R}_1$ and $\mathbf{R}^*_2 = \sigma_\varepsilon^{-2}\mathbf{R}_2$. Then since $(\mathbf{X}_1'\mathbf{V}^{-1}\mathbf{X}_1)^{-1}$ is scalar, $\hat{\beta}^*_1 = (\mathbf{X}_1'\mathbf{V}^{-1}\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{V}^{-1}\mathbf{Y}$ may be written as

$$\hat{\beta}_1^* = \frac{\mathbf{X}_1'\mathbf{V}^{-1}\mathbf{Y}}{\mathbf{X}_1'\mathbf{V}^{-1}\mathbf{X}_1} = \frac{\mathbf{X}_1'\mathbf{PP}'\mathbf{Y}}{\mathbf{X}_1'\mathbf{PP}'\mathbf{X}_1}$$

$$= \frac{\mathbf{X}_1'([\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1}]\mathbf{R}_1 + \sigma_\varepsilon^{-2}\mathbf{R}_2)\mathbf{Y}}{\mathbf{X}_1'([\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1}]\mathbf{R}_1 + \sigma_\varepsilon^{-2}\mathbf{R}_2)\mathbf{X}_1}$$

$$= \frac{\mathbf{X}_1'(\mathbf{R}_1^* + \mathbf{R}_2^*)\mathbf{Y}}{\mathbf{X}_1'(\mathbf{R}_1^* + \mathbf{R}_2^*)\mathbf{X}_1}$$

$$= \frac{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{Y} + \mathbf{X}_1'\mathbf{R}_2^*\mathbf{Y}}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1 + \mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1}.$$

Let $d_1^* = \mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1$ and $d_2^* = \mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1$. Then

$$\hat{\beta}_1^* = \frac{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{Y} + \mathbf{X}_1'\mathbf{R}_2^*\mathbf{Y}}{d_1^* + d_2^*}. \qquad (3)$$

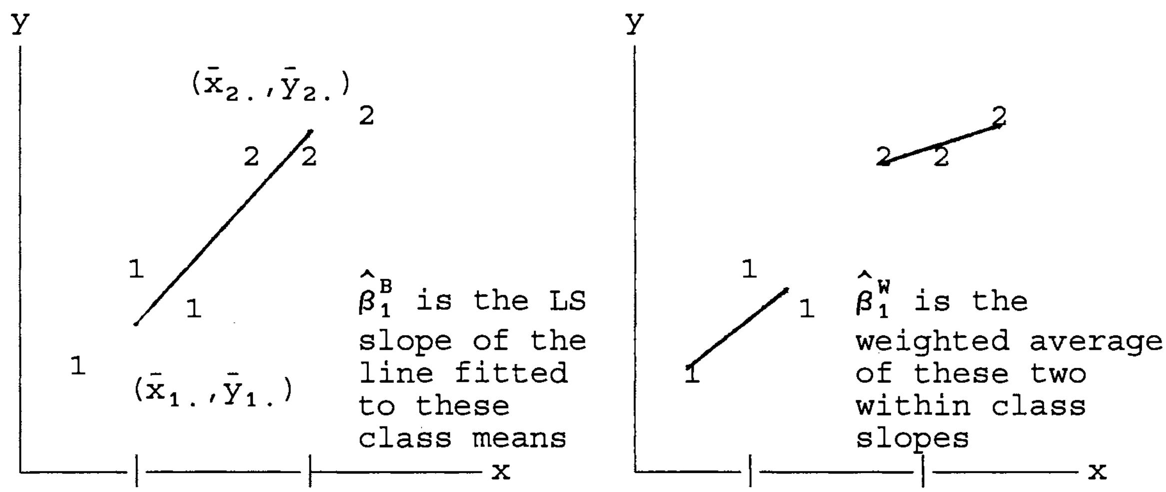
Now to show that $\hat{\beta}_1^*$ is a linear combination of the within and between estimators of β_1, rewrite model (1) as

$$Y_{ijk} = \beta_0 + \beta_1(\bar{x}_{ij.}-\bar{x}_{...}) + \beta_1(x_{ijk}-\bar{x}_{ij.}) + s_i + \varepsilon_{ij}, \qquad (4)$$

where $\bar{x}_{ij.} = b_{ij}^{-1}\sum_{k=1}^{b_{ij}} x_{ijk}$. Then $\hat{\beta}_1^*$ will be a weighted average of two weighted least squares estimators, $\hat{\beta}_1^B$ and $\hat{\beta}_1^W$.

$\hat{\beta}_1^B$ is the WLSE on the set $[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1/2}\mathbf{p}_{1ij}']\mathbf{Y}$ and will be referred to as the weighted least squares between estimator (WLSBE). $\hat{\beta}_1^W$ is the WLSE on the set $[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}']\mathbf{Y}$ and will be referred to as the weighted least squares within estimator (WLSWE).

Now $\hat{\beta}_1^B$ is an estimator of the slope between levels of classification, and $\hat{\beta}_1^W$ is a weighted average of the slope estimators within levels of classification. A simple example of the two estimators with a = 2, r = 1 and b_{11} = b_{21} = 3 is found in Figure 2.



1 = observations from classification 1
2 = observations from classification 2

Figure 2. Simple Example of OLSBE and OLSWE Slope Estimators for a=2, r=1 and b_{11}=b_{21}=3

The WLSBE is the WLSE on $[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1/2}\mathbf{p}_{1ij}']\mathbf{Y}$, thus

$$\hat{\beta}_1^B=\frac{\mathbf{X}_1'[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1/2}\mathbf{p}_{1ij}][\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1/2}\mathbf{p}_{1ij}']\mathbf{Y}}{\mathbf{X}_1'[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1/2}\mathbf{p}_{1ij}][\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1/2}\mathbf{p}_{1ij}']\mathbf{X}}$$

$$=\frac{\mathbf{X}_1'[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1}(\mathbf{I}_{r_i}\otimes b_{ij}^{-1}\mathbf{J}_{b_{ij}})]\mathbf{Y}}{\mathbf{X}_1'[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(b_{ij}\sigma_R^2+\sigma_\varepsilon^2)^{-1}(\mathbf{I}_{r_i}\otimes b_{ij}^{-1}\mathbf{J}_{b_{ij}})]\mathbf{X}_1}$$

$$\hat{\beta}_1^B = \frac{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{Y}}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1} = \frac{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{Y}}{d_1^*}. \tag{5}$$

Since the WLSWE is the WLSE on $[\overset{a}{\underset{i=1}{\oplus}}\overset{r_i}{\underset{j=1}{\oplus}}\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}']\mathbf{Y}$, then

$$\hat{\beta}_1^W = \frac{\mathbf{X}_1'[\overset{a}{\underset{i=1}{\oplus}}\overset{r_i}{\underset{j=1}{\oplus}}\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}][\overset{a}{\underset{i=1}{\oplus}}\overset{r_i}{\underset{j=1}{\oplus}}\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}']\mathbf{Y}}{\mathbf{X}_1'[\overset{a}{\underset{i=1}{\oplus}}\overset{r_i}{\underset{j=1}{\oplus}}\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}][\overset{a}{\underset{i=1}{\oplus}}\overset{r_i}{\underset{j=1}{\oplus}}\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}']\mathbf{X}_1}$$

$$= \frac{\mathbf{X}_1'(\sigma_\varepsilon^{-2}\overset{a}{\underset{i=1}{\oplus}}\overset{r_i}{\underset{j=1}{\oplus}}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_{b_{ij}}-b_{ij}^{-1}\mathbf{J}_{b_{ij}})])\mathbf{Y}}{\mathbf{X}_1'(\sigma_\varepsilon^{-2}\overset{a}{\underset{i=1}{\oplus}}\overset{r_i}{\underset{j=1}{\oplus}}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_{b_{ij}}-b_{ij}^{-1}\mathbf{J}_{b_{ij}})])\mathbf{X}_1}$$

$$= \frac{\mathbf{X}_1'\mathbf{R}_2^*\mathbf{Y}}{\mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1} = \frac{\mathbf{X}_1'\mathbf{R}_2^*\mathbf{Y}}{d_2^*}. \tag{6}$$

Recall the linear combination form of the WLSE from (3),

$$\hat{\beta}_1^* = \frac{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{Y} + \mathbf{X}_1'\mathbf{R}_2^*\mathbf{Y}}{d_1^* + d_2^*}$$

$$= \frac{d_1^*(\mathbf{X}_1'\mathbf{R}_1^*\mathbf{Y}/d_1^*) + d_2^*(\mathbf{X}_1'\mathbf{R}_2^*\mathbf{Y}/d_2^*)}{d_1^* + d_2^*}$$

$$\hat{\beta}_1^* = \frac{d_1^*\hat{\beta}_1^B + d_2^*\hat{\beta}_1^W}{d_1^* + d_2^*}. \tag{7}$$

Thus, the WLSE, $\hat{\beta}_1^*$, for β_1 is a linear combination of the WLSBE, $\hat{\beta}_1^B$, and the WLSWE, $\hat{\beta}_1^W$.

Special Cases

Ordinary Least Squares Estimators

The ordinary least squares estimator, $\hat{\beta}_1$, can be treated as a special case of the weighted least squares estimator in which $\sigma_R^2 = 0$. Then $\mathbf{V} = \sigma_\varepsilon^2\mathbf{I}_n$, $\mathbf{V}^{-1} = \sigma_\varepsilon^{-2}\mathbf{I}_n$, and $\mathbf{P}^*_{ij} = (\sigma_\varepsilon^{-1}\mathbf{p}_{1ij}|\sigma_\varepsilon^{-1}\mathbf{P}_{b_{ij}}) = \sigma_\varepsilon^{-1}(\mathbf{p}_{1ij}|\mathbf{P}_{b_{ij}})$. Then

$$\mathbf{P}^*_{ij}\mathbf{P}^{*\prime}_{ij} = \sigma_\varepsilon^{-1}(\mathbf{p}_{1ij}|\mathbf{P}_{b_{ij}})\sigma_\varepsilon^{-1}\begin{pmatrix}\mathbf{p}_{1ij}' \\ \mathbf{P}_{b_{ij}}'\end{pmatrix}$$

$$= \sigma_\varepsilon^{-2}(\mathbf{p}_{1ij}\mathbf{p}_{1ij}' + \mathbf{P}_{b_{ij}}\mathbf{P}_{b_{ij}}')$$

$$= \sigma_\varepsilon^{-2}[b_{ij}^{-1/2}\mathbf{1}_{b_{ij}}b_{ij}^{-1/2}\mathbf{1}_{b_{ij}}' + (\mathbf{I}_{b_{ij}} - b_{ij}^{-1}\mathbf{J}_{b_{ij}})]$$

$$= \sigma_\varepsilon^{-2}[b_{ij}^{-1}\mathbf{J}_{b_{ij}} + (\mathbf{I}_{b_{ij}} - b_{ij}^{-1}\mathbf{J}_{b_{ij}})]$$ and

and $$\mathbf{PP}' = \begin{pmatrix} \mathbf{P}^*_{ij}\mathbf{P}^{*\prime}_{ij} & \mathbf{0} & . & . & \mathbf{0} \\ \mathbf{0} & \mathbf{P}^*_{ij}\mathbf{P}^{*\prime}_{ij} & \mathbf{0} & . & . \\ . & & . & & . \\ . & & & . & \\ \mathbf{0} & & . & . \ \mathbf{0} & \mathbf{P}^*_{ij}\mathbf{P}^{*\prime}_{ij} \end{pmatrix}$$

$$= \bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}\mathbf{P}^*_{ij}\mathbf{P}^{*\prime}_{ij}$$

$$\mathbf{PP}' = \sigma_\varepsilon^{-2}\Big(\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(\mathbf{I}_{r_i}\otimes b_{ij}^{-1}\mathbf{J}_{b_{ij}}) + \bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_{b_{ij}}-b_{ij}^{-1}\mathbf{J}_{b_{ij}})]\Big)$$

$$= \sigma_\varepsilon^{-2}\mathbf{R}_1 + \sigma_\varepsilon^{-2}\mathbf{R}_2$$

$$= \sigma_\varepsilon^{-2}\mathbf{I}_n.$$ Then

$$\hat{\beta}_1 = \frac{\mathbf{X}_1'\mathbf{Y}}{\mathbf{X}_1'\mathbf{X}_1} = \frac{\mathbf{X}_1'\mathbf{PP}'\mathbf{Y}}{\mathbf{X}_1'\mathbf{PP}'\mathbf{X}_1}$$

$$= \frac{\sigma_\varepsilon^{-2}\mathbf{X}_1'(\mathbf{R}_1 + \mathbf{R}_2)\mathbf{Y}}{\sigma_\varepsilon^{-2}\mathbf{X}_1'(\mathbf{R}_1 + \mathbf{R}_2)\mathbf{X}_1}$$

$$\hat{\beta}_1 = \frac{\mathbf{X}_1'\mathbf{R}_1\mathbf{Y} + \mathbf{X}_1'\mathbf{R}_2\mathbf{Y}}{\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1 + \mathbf{X}_1'\mathbf{R}_2\mathbf{X}_1}$$

$$= \frac{\mathbf{X}_1'\mathbf{R}_1\mathbf{Y} + \mathbf{X}_1'\mathbf{R}_2\mathbf{Y}}{d_1 + d_2} \qquad (8)$$

where $d_1 = \mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1$ and $d_2 = \mathbf{X}_1'\mathbf{R}_2\mathbf{X}_1$. Similarly, the OLSBE is

$$\hat{\beta}_1^B = \frac{\mathbf{X}_1'[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(\mathbf{I}_{r_i}\otimes b_{ij}^{-1}\mathbf{J}_{b_{ij}})]\mathbf{Y}}{\mathbf{X}_1'[\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(\mathbf{I}_{r_i}\otimes b_{ij}^{-1}\mathbf{J}_{b_{ij}})]\mathbf{X}_1}$$

$$= \frac{\mathbf{X}_1'\mathbf{R}_1\mathbf{Y}}{\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1} = \frac{\mathbf{X}_1'\mathbf{R}_1\mathbf{Y}}{d_1} \qquad (9)$$

and the OLSWE is

$$\hat{\beta}_1^W = \frac{\mathbf{X}_1'(\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_{b_{ij}}-b_{ij}^{-1}\mathbf{J}_{b_{ij}})])\mathbf{Y}}{\mathbf{X}_1'(\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_{b_{ij}}-b_{ij}^{-1}\mathbf{J}_{b_{ij}})])\mathbf{X}_1}$$

$$= \frac{\mathbf{X}_1'\mathbf{R}_2\mathbf{Y}}{\mathbf{X}_1'\mathbf{R}_2\mathbf{X}_1} = \frac{\mathbf{X}_1'\mathbf{R}_2\mathbf{Y}}{d_2}. \qquad (10)$$

Then from (8), the OLSE is a linear combination of the OLSBE and the OLSWE:

$$\hat{\beta}_1 = \frac{\mathbf{X}_1'\mathbf{R}_1\mathbf{Y} + \mathbf{X}_1'\mathbf{R}_2\mathbf{Y}}{d_1 + d_2}$$

$$= \frac{d_1(\mathbf{X}_1'\mathbf{R}_1\mathbf{Y}/d_1) + d_2(\mathbf{X}_1'\mathbf{R}_2\mathbf{Y}/d_2)}{d_1 + d_2}$$

$$= \frac{d_1\hat{\beta}_1^B + d_2\hat{\beta}_1^W}{d_1 + d_2}. \qquad (11)$$

Note that the variance component cancels out of the WLSWE so the WLSWE and the OLSWE are equal.

Balanced Designs

The forms of the estimators are simplified when the design is balanced over replications, over time or over both replications and time. Appropriate $\mathbf{R}_1$ and $\mathbf{R}_2$ matrices may be defined for each situation and these matrices substituted into the previous equations for $\hat{\beta}_1^*$, $\hat{\beta}_1^B$, $\hat{\beta}_1^W$, and $\hat{\beta}_1$ to obtain the estimators of β_1.

First assume the design is balanced in r so that $r_i=r$ for all classification levels and is unbalanced over levels of time so that $b_{ij} \neq b_{i'j'}$ for some $i \neq i'$ or $j \neq j'$. Then $\mathbf{R}_1=\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r} b_{ij}^{-1}\mathbf{J}_{b_{ij}}$ and $\mathbf{R}_2=\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r}(\mathbf{I}_{b_{ij}}-b_{ij}^{-1}\mathbf{J}_{b_{ij}})$ for $i=1, \ldots, a$ and $j = 1, \ldots, r$.

Now assume the design is balanced over levels of time so that $b_{ij} = b$ for all $i = 1, \ldots, a$ and $j = 1, \ldots, r$, but is not balanced over levels of replication so that $r_i \neq r_{i'}$ for some $i \neq i'$. Then $\mathbf{R}_1 = \bigoplus_{i=1}^{a}(\mathbf{I}_{r_i}\otimes b^{-1}\mathbf{J}_b)$ and $\mathbf{R}_2 = \bigoplus_{i=1}^{a}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_b-b^{-1}\mathbf{J}_b)]$.

Finally, assume that the design is balanced over both levels of time and replication so that $r_i = r$ and $b_{ij} = b$ for $i = 1, \ldots, a$ and $j = 1, \ldots, r$. Then $\mathbf{R}_1 = \mathbf{I}_a\otimes\mathbf{I}_r\otimes b^{-1}\mathbf{J}_b$, and $\mathbf{R}_2 = \mathbf{I}_a\otimes\mathbf{I}_r\otimes(\mathbf{I}_b-b^{-1}\mathbf{J}_b)$.

When the design is completely balanced, $\mathbf{V}^{-1} = (b\sigma_R^2+\sigma_\varepsilon^2)^{-1}(\mathbf{I}_a\otimes\mathbf{I}_r\otimes b^{-1}\mathbf{J}_b) + \sigma_\varepsilon^{-2}[\mathbf{I}_a\otimes\mathbf{I}_r\otimes(\mathbf{I}_b-b^{-1}\mathbf{J}_b)]$, hence $\mathbf{PP}' = (b\sigma_R^2+\sigma_\varepsilon^2)^{-1}\mathbf{R}_1 + \sigma_\varepsilon^{-2}\mathbf{R}_2$.

The form of the WLSE estimator is then

$$\hat{\beta}_1^* = \frac{\mathbf{X}_1'[(b\sigma_R^2+\sigma_\varepsilon^2)^{-1}\mathbf{R}_1 + \sigma_\varepsilon^{-2}\mathbf{R}_2]\mathbf{Y}}{\mathbf{X}_1'[(b\sigma_R^2+\sigma_\varepsilon^2)^{-1}\mathbf{R}_1 + \sigma_\varepsilon^{-2}\mathbf{R}_2]\mathbf{X}_1}$$

$$= \frac{(b\sigma_R^2+\sigma_\varepsilon^2)^{-1}\mathbf{X}_1'\mathbf{R}_1\mathbf{Y} + \sigma_\varepsilon^{-2}\mathbf{X}_1'\mathbf{R}_2\mathbf{Y}}{(b\sigma_R^2+\sigma_\varepsilon^2)^{-1}\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1 + \sigma_\varepsilon^{-2}\mathbf{X}_1'\mathbf{R}_2\mathbf{X}_1}$$

$$= \frac{d_3\mathbf{X}_1'\mathbf{R}_1\mathbf{Y} + d_4\mathbf{X}_1'\mathbf{R}_2\mathbf{Y}}{d_3d_1 + d_4d_2}$$

$$= \frac{d_3d_1\hat{\beta}_1^B + d_4d_2\hat{\beta}_1^W}{d_3d_1 + d_4d_2} \qquad (12)$$

where $d_3 = (b\sigma_R^2+\sigma_\varepsilon^2)^{-1}$ and $d_4 = \sigma_\varepsilon^{-2}$. Thus when the design is balanced, the WLSE estimator is a weighted average of the OLSBE and the OLSWE.

Lack of Replication

If there is no replication in the experiment under consideration, i.e. r = 1 for all classifications, then the model, as well as the forms of the $\mathbf{X}_1$, $\mathbf{Y}$ and $\mathbf{V}$ matrices are simplified, as are the estimators for β_1. The centered model is now

$$Y_{ij} = \beta_0 + \beta_1(x_{ij}-\bar{x}_{..}) + s_i + \varepsilon_{ij} \qquad (13)$$

where $\bar{x}_{..}$ is the overall mean of the x_{ij}'s, and ε_{ij} is the random error such that $E(\varepsilon_{ij}) = 0$ for all i = 1, ..., *a*, and j = 1, ... , b_i. Then $\sum_{i=1}^{a} b_i = n$, where n is the total number of observations.

The matrix form of the model remains the same as (2). However, now $\mathbf{X}_1$ is an *nx1* vector such that $\mathbf{X}_1 = (x_{11}-\bar{x}_{..},$

$\ldots, x_{1b_1}-\bar{x}_{..}, \ldots, x_{a1}-\bar{x}_{..}, \ldots, x_{ab_a}-\bar{x}_{..})'$ and $\mathbf{Y} = (y_{11}, \ldots, y_{1b_i}, \ldots, y_{a1}, \ldots, y_{ab_i})'$. The covariance matrix is now $\mathbf{V} = \sigma^2_{R_i}\bigoplus_{i=1}^{a}\mathbf{J}_{b_i} + \sigma^2_{\varepsilon}\mathbf{I}_n$, therefore $\mathbf{V}^{-1} = \bigoplus_{i=1}^{a}(b_i\sigma^2_R+\sigma^2_{\varepsilon})^{-1}b_i^{-1}\mathbf{J}_{b_i} + \sigma_{\varepsilon}^{-2}\bigoplus_{i=1}^{a}(\mathbf{I}_{b_i}-b_i^{-1}\mathbf{J}_{b_i})$. Then the matrices $\mathbf{R}_1$ and $\mathbf{R}_2$ are now defined as $\mathbf{R}_1 = \bigoplus_{i=1}^{a}b_i^{-1}\mathbf{J}_{b_i}$ and $\mathbf{R}_2 = \bigoplus_{i=1}^{a}(\mathbf{I}_{b_i}-b_i^{-1}\mathbf{J}_{b_i})$.

If the design with r = 1 is balanced over levels of time so that $b_i = b$ for all $i = 1, \ldots, a$, then all matrices are simply Kronecker products. Hence, $\mathbf{V} = \sigma^2_R(\mathbf{I}_a\otimes\mathbf{J}_b) + \sigma^2_{\varepsilon}(\mathbf{I}_a\otimes\mathbf{I}_b)$, $\mathbf{R}_1 = \mathbf{I}_a\otimes b^{-1}\mathbf{J}_b$, and $\mathbf{R}_2 = \mathbf{I}_a\otimes(\mathbf{I}_b-b^{-1}\mathbf{J}_b)$.

CHAPTER III

QUALITY OF THE LEAST SQUARES ESTIMATORS

A desirable property of the ordinary and weighted least squares estimators of β_1 is that they are unbiased when the model is assumed to be known. In fact, if there is no significant bias, so that $s_i = 0$ for all $i = 1, \ldots, a$, the WLSE is the best linear unbiased estimator (BLUE) of β_1 for the layout under consideration when the covariance matrix is $\Sigma = \mathbf{V} = \sigma_R^2[\oplus_{i=1}^{a}\oplus_{j=1}^{r_i}(\mathbf{I}_{r_i}\otimes\mathbf{J}_{b_{ij}})] + \sigma_\varepsilon^2\mathbf{I}_n$. If $\sigma_R^2 = 0$, then $\mathbf{V}$ simplifies to $\sigma_\varepsilon^2\mathbf{I}_n$, and the OLSE is BLUE for β_1.

Effect of Non-zero Constant Bias on the Estimators

Assume now that $s_i \neq 0$, but is constant across levels of classification so that $s_i = c$ for all $i = 1, \ldots, a$. Then $E(\mathbf{Y}) = \mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)}$ where $\mathbf{X}_2\beta^{(2)} = c\oplus_{i=1}^{a}\oplus_{j=1}^{r_i}(\mathbf{1}_{r_i}\otimes\mathbf{1}_{b_{ij}})$. Note

$$\mathbf{R}_1\mathbf{X}_2\beta^{(2)} = \oplus_{i=1}^{a}\oplus_{j=1}^{r_i}(\mathbf{I}_{r_i}\otimes b_{ij}^{-1}\mathbf{J}_{b_{ij}})\, c\oplus_{i=1}^{a}\oplus_{j=1}^{r_i}(\mathbf{1}_{r_i}\otimes\mathbf{1}_{b_{ij}}) = \mathbf{X}_2\beta^{(2)} \text{ and}$$

$$\mathbf{R}_2\mathbf{X}_2\beta^{(2)} = \oplus_{i=1}^{a}\oplus_{j=1}^{r_i}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_{b_{ij}} - b_{ij}^{-1}\mathbf{J}_{b_{ij}})]\, c\oplus_{i=1}^{a}\oplus_{j=1}^{r_i}(\mathbf{1}_{r_i}\otimes\mathbf{1}_{b_{ij}}) = \mathbf{0}.$$

Similarly, $\mathbf{R}_1^*\mathbf{X}_2\beta^{(2)} = \mathbf{X}_2\beta^{(2)}$ and $\mathbf{R}_2^*\mathbf{X}_2\beta^{(2)} = \mathbf{0}$. Also, $\mathbf{X}_1'\mathbf{X}_2\beta^{(2)} = 0$. Then the WLSBE is an unbiased estimator of β_1,

$$E(\hat{\beta}_1^B) = \left[\frac{\mathbf{X}_1'\mathbf{R}_1^*}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1}\right]E(\mathbf{Y})$$

$$E(\hat{\beta}_1^B) = \left[\frac{\mathbf{X}_1'\mathbf{R}_1^*}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1}\right](\mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= \beta_1 + \left[\frac{\mathbf{X}_1'\mathbf{R}_1^*}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1}\right]\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + \left[\frac{\mathbf{X}_1'}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1}\right]\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + (\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)} = \beta_1.$$

Since the OLSBE is of the same form as the WLSBE with $\mathbf{R}_1^*$ replaced by $\mathbf{R}_1$, the OLSBE is also unbiased for β_1 when bias is constant across classifications.

Similarly, the WLSWE, hence the OLSWE, is unbiased for the β_1 slope parameter,

$$E(\hat{\beta}_1^W) = \left[\frac{\mathbf{X}_1'\mathbf{R}_2^*}{\mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1}\right]E(\mathbf{Y})$$

$$= \left[\frac{\mathbf{X}_1'\mathbf{R}_2^*}{\mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1}\right](\mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= \beta_1 + \left[\frac{\mathbf{X}_1'\mathbf{R}_2^*}{\mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1}\right]\mathbf{X}_2\beta^{(2)} = \beta_1.$$

The WLSE estimator is unbiased for β_1 since it is a linear combination of unbiased estimators,

$$E(\hat{\beta}_1^*) = \left[\frac{d_1^*}{d_1^* + d_2^*}\right] E(\hat{\beta}_1^B) + \left[\frac{d_2^*}{d_1^* + d_2^*}\right] E(\hat{\beta}_1^W)$$

$$= \left[\frac{d_1^*}{d_1^* + d_2^*}\right] \beta_1 + \left[\frac{d_2^*}{d_1^* + d_2^*}\right] \beta_1$$

$$= \beta_1 .$$

Finally, when $s_i = c$, the OLSE is also an unbiased estimator of β_1,

$$E(\hat{\beta}_1) = \frac{d_1\hat{\beta}_1^B + d_2\hat{\beta}_1^W}{d_1 + d_2}$$

$$= \left[\frac{d_1}{d_1 + d_2}\right] E(\hat{\beta}_1^B) + \left[\frac{d_2}{d_1 + d_2}\right] E(\hat{\beta}_1^W)$$

$$= \left[\frac{d_1}{d_1 + d_2}\right] \beta_1 + \left[\frac{d_2}{d_1 + d_2}\right] \beta_1$$

$$= \beta_1 .$$

Effect of Non-constant Bias on the Estimators

If the bias in the model is not constant, but changes over the levels of classification so that $s_i \neq s_{i'}$ for some $i \neq i'$, all of the least squares estimators except the within classifications estimator become biased for β_1. The expectation for $\mathbf{Y}$ is still $E(\mathbf{Y}) = \mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)}$, but now

$\mathbf{X}_2\beta^{(2)} = \bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(s_i\otimes\mathbf{1}_{r_i}\otimes\mathbf{1}_{b_{ij}})$. Note

$\mathbf{R}_1\mathbf{X}_2\beta^{(2)} = \bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(\mathbf{I}_{r_i}\otimes b_{ij}^{-1}\mathbf{J}_{b_{ij}})\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(s_i\otimes\mathbf{1}_{r_i}\otimes\mathbf{1}_{b_{ij}}) = \mathbf{X}_2\beta^{(2)}$ and

$\mathbf{R}_2\mathbf{X}_2\beta^{(2)} = \bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}[\mathbf{I}_{r_i}\otimes(\mathbf{I}_{b_{ij}} - b_{ij}^{-1}\mathbf{J}_{b_{ij}})]\bigoplus_{i=1}^{a}\bigoplus_{j=1}^{r_i}(s_i\otimes\mathbf{1}_{r_i}\otimes\mathbf{1}_{b_{ij}}) = \mathbf{0}$.

Then WLSBE is no longer an unbiased estimator of β_1,

$$E(\hat{\beta}_1^B) = \left[\frac{\mathbf{X}_1'\mathbf{R}_1^*}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1}\right]E(\mathbf{Y})$$

$$= \left[\frac{\mathbf{X}_1'\mathbf{R}_1^*}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1}\right](\mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= \beta_1 + \left[\frac{\mathbf{X}_1'\mathbf{R}_1^*}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1}\right]\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + \left[\frac{\mathbf{X}_1'}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1}\right]\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + (\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}.$$

Since the OLSBE is of the same form as the WLSBE with $\mathbf{R}_1^*$ replaced by $\mathbf{R}_1$, the OLSBE is also becomes biased for β_1 when bias is not constant across classifications.

However, the WLSWE/OLSWE remains unbiased for β_1 when bias is not constant,

$$E(\hat{\beta}_1^W) = \left[\frac{\mathbf{X}_1'\mathbf{R}_2^*}{\mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1}\right]E(\mathbf{Y})$$

$$E(\hat{\beta}_1^W) = \left[\frac{\mathbf{X}_1'\mathbf{R}_2^*}{\mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1}\right](\mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= \beta_1 + \left[\frac{\mathbf{X}_1'\mathbf{R}_2^*}{\mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1}\right]\mathbf{X}_2\beta^{(2)} = \beta_1 .$$

The WLSE estimator is biased for β_1 since it is a linear combination of the WLSBE and WLSWE,

$$E(\hat{\beta}_1^*) = \left[\frac{d_1^*}{d_1^* + d_2^*}\right]E(\hat{\beta}_1^B) + \left[\frac{d_2^*}{d_1^* + d_2^*}\right]E(\hat{\beta}_1^W)$$

$$= \left[\frac{d_1^*}{d_1^* + d_2^*}\right](\beta_1 + (\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}) + \left[\frac{d_2^*}{d_1^* + d_2^*}\right]\beta_1$$

$$= \beta_1 + \left[\frac{d_1^*}{d_1^* + d_2^*}\right](\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + \left[\frac{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1}{\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1 + \mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1}\right](\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + (\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1 + \mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1)^{-1}(\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1)(\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + (\mathbf{X}_1'\mathbf{R}_1^*\mathbf{X}_1 + \mathbf{X}_1'\mathbf{R}_2^*\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)} .$$

Since the OLSE is a linear combination of the OLSBE and OLSWE estimators of β_1, $\hat{\beta}_1$ is also biased when $s_i \neq s_{i'}$, for some $i \neq i'$,

$$E(\hat{\beta}_1) = \frac{d_1\hat{\beta}_1^B + d_2\hat{\beta}_1^W}{d_1 + d_2}$$

$$= \left[\frac{d_1}{d_1 + d_2}\right] E(\hat{\beta}_1^B) + \left[\frac{d_2}{d_1 + d_2}\right] E(\hat{\beta}_1^W)$$

$$= \left[\frac{d_1}{d_1 + d_2}\right] (\beta_1 + (\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}) + \left[\frac{d_2}{d_1 + d_2}\right] \beta_1$$

$$= \beta_1 + \left[\frac{d_1}{d_1 + d_2}\right] (\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + \left[\frac{\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1}{\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1 + \mathbf{X}_1'\mathbf{R}_2\mathbf{X}_1}\right] (\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + \left[\frac{\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1}{\mathbf{X}_1'\mathbf{X}_1}\right] (\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}.$$

Therefore, the ordinary least squares within estimator is always unbiased for β_1. The WLSE, WLSBE, OLSBE and OLSE all become biased for β_1 when bias is not constant across the levels of classification.

CHAPTER IV

DERIVATION OF VARIANCE ESTIMATORS

In Chapters II and III, the weighted least squares estimators are functions of the unknown parameters σ_R^2 and σ_ε^2. It is of interest then to find estimators $\hat{\sigma}_R^2$ and $\hat{\sigma}_\varepsilon^2$. The derivation of these variance estimators may be accomplished by using an analysis of variance (ANOVA) table and finding the expected mean squares (EMS) of the between and within residuals. The tables will only be produced for the balanced designs but can be adapted for the unbalanced designs with the appropriate changes in the sum of squares matrices. First consider the case with replication, since replication is necessary to find an estimator of σ_R^2 when the bias term is unknown and bias is non-constant.

The source column of the ANOVA table, Table I on the following page, is divided into between classifications and within classifications subheadings with further subdivisions for regression and residual. The residual for the between classifications is then divided into a lack of fit (LOF) portion and a pure error (PE) portion.

The sum of squares for each source is designated by $\mathbf{Y}'\mathbf{A}_i\mathbf{Y}$ for each of the i = 1, ..., 7 lines in the ANOVA table, and each $\mathbf{A}_i$ is called a sum of squares matrix. The $\mathbf{A}_1$ matrix is the sum of squares matrix for the overall mean, $\mathbf{A}_2$ for between regression, $\mathbf{A}_3$ for between residual, $\mathbf{A}_4$ for lack of fit for between residual, $\mathbf{A}_5$ for pure error for between residual, $\mathbf{A}_6$ for within regression and $\mathbf{A}_7$ for within residual. The degrees of freedom for each source are equal to the trace of the corresponding sum of squares matrix, since each $\mathbf{A}_i$ is idempotent.

Since the ANOVA table is divided into between and within sources of variation, the original $\mathbf{X}_1$ matrix is divided into between and within components for use in the sum of squares matrices. Let the *arx1* matrix $\mathbf{X}_B^* =$ $(\bar{x}_{11.}-\bar{x}_{...}, \ldots, \bar{x}_{1r.}-\bar{x}_{...}, \ldots, \bar{x}_{a1.}-\bar{x}_{...}, \ldots, \bar{x}_{ar.}-\bar{x}_{...})'$ and the *arbx1* matrix $\mathbf{X}_B = \mathbf{X}_B^* \otimes \mathbf{1}_b$. Now $(\mathbf{X}_B'\mathbf{X}_B)^{-1} =$ $(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1} \otimes b^{-1}$ so that $\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B' = \mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime} \otimes b^{-1}\mathbf{J}_b$. In addition, the relationships $(\mathbf{1}_a' \otimes \mathbf{1}_r')\mathbf{X}_B^* = 0$ (a scalar) and $(a^{-1}\mathbf{J}_a \otimes r^{-1}\mathbf{J}_r)\mathbf{X}_B^* = \mathbf{0}$ (of dimension *ar*x1) will prove useful.

Table I. Analysis of Variance Table With Replication

Source	df	Sum of Squares
Between	ar	$\mathbf{Y}'[\mathbf{I}_a \otimes \mathbf{I}_r \otimes b^{-1}\mathbf{J}_b]\mathbf{Y}$
Mean	1	$\mathbf{Y}'\mathbf{A}_1\mathbf{Y}$
Regression	1	$\mathbf{Y}'\mathbf{A}_2\mathbf{Y}$
Residual	ar-2	$\mathbf{Y}'\mathbf{A}_3\mathbf{Y}$
LOF	a-2	$\mathbf{Y}'\mathbf{A}_4\mathbf{Y}$
PE	a(r-1)	$\mathbf{Y}'\mathbf{A}_5\mathbf{Y}$
Within	ar(b-1)	$\mathbf{Y}'[\mathbf{I}_a \otimes \mathbf{I}_r \otimes (\mathbf{I}_b - b^{-1}\mathbf{J}_b)]\mathbf{Y}$
Regression	1	$\mathbf{Y}'\mathbf{A}_6\mathbf{Y}$
Residual	ar(b-1)-1	$\mathbf{Y}'\mathbf{A}_7\mathbf{Y}$
Total	ab	$\mathbf{Y}'[\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b]\mathbf{Y}$

Also let $\mathbf{X}_W = (x_{111}-\bar{x}_{11.}, \ldots, x_{1rb}-\bar{x}_{1r.}, \ldots, x_{a11}-\bar{x}_{a1.}, \ldots, x_{arb}-\bar{x}_{ar..})'$. Then $\mathbf{X}_B + \mathbf{X}_W = \mathbf{X}_1$. Note the relationships $(\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{1}_b')\mathbf{X}_W = \mathbf{0}$ (of dimension *arx1*) and $(\mathbf{I}_a \otimes \mathbf{I}_r \otimes b^{-1}\mathbf{J}_b)\mathbf{X}_W = \mathbf{0}$ (of dimension *arbx1*).

Define the *rx1* matrix $\mathbf{B}_i = (\bar{x}_{11.}-\bar{x}_{1...}, \ldots, \bar{x}_{1r.}-\bar{x}_{1..})'$ for i = 1, ..., *a*. This matrix will be used to construct the sum of squares matrix for pure error.

Then $\mathbf{A}_1=[a^{-1}\mathbf{J}_a \otimes r^{-1}\mathbf{J}_r \otimes b^{-1}\mathbf{J}_b]$, $\mathbf{A}_2=[\mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime} \otimes b^{-1}\mathbf{J}_b]$, $\mathbf{A}_3=[\{(\mathbf{I}_a \otimes \mathbf{I}_r)-(a^{-1}\mathbf{J}_a \otimes r^{-1}\mathbf{J}_r)-(\mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime})\} \otimes b^{-1}\mathbf{J}_b]$, $\mathbf{A}_4=[(\{(\mathbf{I}_a-a^{-1}\mathbf{J}_a)\otimes r^{-1}\mathbf{J}_r\}-\mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime}-\bigoplus_{i=1}^{a}\mathbf{B}_i(\mathbf{B}_i'\mathbf{B}_i)^{-1}\mathbf{B}_i') \otimes b^{-1}\mathbf{J}_b]$, $\mathbf{A}_5=[(\{\mathbf{I}_a \otimes (\mathbf{I}_r-r^{-1}\mathbf{J}_r)\}-\bigoplus_{i=1}^{a}\mathbf{B}_i(\mathbf{B}_i'\mathbf{B}_i)^{-1}\mathbf{B}_i') \otimes b^{-1}\mathbf{J}_b]$, $\mathbf{A}_6=[\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W']$, and $\mathbf{A}_7=[(\mathbf{I}_a \otimes \mathbf{I}_r \otimes [\mathbf{I}_b-b^{-1}\mathbf{J}_b])-\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W']$. One may notice that the sum of squares matrix for between classifications is equal to the $\mathbf{R}_1$ matrix used in the ordinary least squares between estimator, and the sum of squares matrix for within classifications is equal to the $\mathbf{R}_2$ matrix used in the ordinary least squares within estimator.

The expected mean square of $\mathbf{Y}'\mathbf{A}_i\mathbf{Y}$ is

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_i\mathbf{Y}}{tr(\mathbf{A}_i)}\right] = \frac{tr(\mathbf{A}_i\Sigma) + \mu'\mathbf{A}_i\mu}{tr(\mathbf{A}_i)}$$

where $tr(\mathbf{A}_i)$ is the trace of the matrix $\mathbf{A}_i$, μ is the mean vector for **Y**, where $\mu=\mathbf{X}_B\beta_1+\mathbf{X}_2\beta^{(2)}$ for the between component and $\mu=\mathbf{X}_W\beta_1+\mathbf{X}_2\beta^{(2)}$ for the within component, and Σ is the covariance matrix of **Y**. Calculations for expected mean squares can be found in Appendix A.

Assuming $\Sigma = \sigma_R^2(\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{J}_b) + \sigma_\varepsilon^2(\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b)$, the expected mean squares for between classifications pure error and the within classifications residual are:

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_5\mathbf{Y}}{tr(\mathbf{A}_5)}\right] = E\left[\frac{\mathbf{Y}'[(\{\mathbf{I}_a \otimes (\mathbf{I}_r - r^{-1}\mathbf{J}_r)\} - \bigoplus_{i=1}^{a} \mathbf{B}_i(\mathbf{B}_i'\mathbf{B}_i)^{-1}\mathbf{B}_i') \otimes b^{-1}\mathbf{J}_b]\mathbf{Y}}{a(r-1)}\right]$$

$$= \sigma_\varepsilon^2 + b\sigma_R^2, \text{ and}$$

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_7\mathbf{Y}}{tr(\mathbf{A}_7)}\right] = E\left[\frac{\mathbf{Y}'[(\mathbf{I}_a \otimes \mathbf{I}_r \otimes [\mathbf{I}_b - b^{-1}\mathbf{J}_b]) - \mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W']\mathbf{Y}}{ar(b-1)-1}\right]$$

$$= \sigma_\varepsilon^2, \text{ respectively.}$$

These expected mean squares hold for both constant and non-constant bias.

Then unbiased estimators of σ_ε^2 and σ_R^2 are

$$\hat{\sigma}_\varepsilon^2 = \frac{\mathbf{Y}'\mathbf{A}_7\mathbf{Y}}{tr(\mathbf{A}_7)}$$

$$= \frac{\mathbf{Y}'[(\mathbf{I}_a \otimes \mathbf{I}_r \otimes [\mathbf{I}_b - b^{-1}\mathbf{J}_b]) - \mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W']\mathbf{Y}}{ar(b-1)-1}, \text{ and} \tag{14}$$

$$\hat{\sigma}_R^2 = b^{-1}\left[\frac{\mathbf{Y}'\mathbf{A}_5\mathbf{Y}}{tr(\mathbf{A}_5)} - \frac{\mathbf{Y}'\mathbf{A}_7\mathbf{Y}}{tr(\mathbf{A}_7)}\right]$$

$$= b^{-1}\left[\frac{\mathbf{Y}'[(\{\mathbf{I}_a \otimes (\mathbf{I}_r - r^{-1}\mathbf{J}_r)\} - \bigoplus_{i=1}^{a} \mathbf{B}_i(\mathbf{B}_i'\mathbf{B}_i)^{-1}\mathbf{B}_i') \otimes b^{-1}\mathbf{J}_b]\mathbf{Y}}{a(r-1)}\right]$$

$$- b^{-1}\left[\frac{\mathbf{Y}'[(\mathbf{I}_a \otimes \mathbf{I}_r \otimes [\mathbf{I}_b - b^{-1}\mathbf{J}_b]) - \mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W']\mathbf{Y}}{ar(b-1)-1}\right]. \tag{15}$$

If $(ar(b-1)-1)\mathbf{Y}'\mathbf{A}_5\mathbf{Y} > (a(r-1))\mathbf{Y}'\mathbf{A}_7\mathbf{Y}$ then $\hat{\sigma}^2_R$ as constructed from the ANOVA table will be negative. If this occurs, set $\hat{\sigma}^2_R$ equal to zero and use a pooled estimator for σ^2_ε, that is,

$$\hat{\sigma}^2_{\varepsilon p} = \frac{(a(r-1))\mathbf{Y}'\mathbf{A}_5\mathbf{Y} + (ar(b-1)-1)\mathbf{Y}'\mathbf{A}_7\mathbf{Y}}{a(r-1) + (ar(b-1)-1)}. \qquad (16)$$

Special Cases

Constant Bias

If bias is constant across the levels of classification so that $s_i=c$ for all $i = 1, \ldots, a$, then $\beta^{(2)\prime}\mathbf{X}_2' = c(\mathbf{1}_a'\otimes\mathbf{1}_r'\otimes\mathbf{1}_b')$, so $\beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_B = 0$. Hence, the EMS of the between classifications residual has no bias term:

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_3\mathbf{Y}}{tr(\mathbf{A}_3)}\right] = \sigma^2_\varepsilon + b\sigma^2_R + (\beta^{(2)\prime}\mathbf{X}_2'\mathbf{A}_3\mathbf{X}_2\beta^{(2)})/(ar-2)$$

$$= \sigma^2_\varepsilon + b\sigma^2_R + \frac{\beta^{(2)\prime}\mathbf{X}_2'[\{(\mathbf{I}_a\otimes\mathbf{I}_r)-(a^{-1}\mathbf{J}_a\otimes r^{-1}\mathbf{J}_r)\}\otimes b^{-1}\mathbf{J}_b]\mathbf{X}_2\beta^{(2)}}{ar-2}$$
$$- \frac{\beta^{(2)\prime}\mathbf{X}_2'[(\mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime}\otimes b^{-1}\mathbf{J}_b]\mathbf{X}_2\beta^{(2)}}{ar-2}$$

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_3\mathbf{Y}}{tr(\mathbf{A}_3)}\right] = \sigma^2_\varepsilon + b\sigma^2_R.$$

Therefore, an unbiased estimator of σ^2_R can be constructed without dividing this residual into lack of fit and pure error. This implies that replications are unnecessary if bias is constant across levels of classification. If replications are not necessary, the $\mathbf{A}_i$ matrices can be reduced to dimension ***abxab*** and the ANOVA table simplified, as shown in Table II.

Let $\mathbf{A}_8=[a^{-1}\mathbf{J}_a \otimes b^{-1}\mathbf{J}_b]$, $\mathbf{A}_9=[\mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime} \otimes b^{-1}\mathbf{J}_b]$, $\mathbf{A}_{10}=[(\mathbf{I}_a-a^{-1}\mathbf{J}_a-\mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime} \otimes b^{-1}\mathbf{J}_b]$, $\mathbf{A}_{11}=[\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W']$, and $\mathbf{A}_{12}=[(\mathbf{I}_a \otimes [\mathbf{I}_b-b^{-1}\mathbf{J}_b])-\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W']$.

The expected mean squares for residuals for between and within classifications when the bias is constant may now be utilized to find unbiased estimators of σ_R^2 and σ_ε^2. Since

$$E\left(\frac{\mathbf{Y}'\mathbf{A}_{12}\mathbf{Y}}{\mathrm{tr}(\mathbf{A}_{12})}\right) = E\left(\frac{\mathbf{Y}'[(\mathbf{I}_a \otimes [\mathbf{I}_b-b^{-1}\mathbf{J}_b])-\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W']\mathbf{Y}}{a(b-1)-1}\right) = \sigma_\varepsilon^2,$$

an unbiased estimator of σ_ε^2 is

$$\hat{\sigma}_\varepsilon^2 = \left(\frac{\mathbf{Y}'\mathbf{A}_{12}\mathbf{Y}}{\mathrm{tr}(\mathbf{A}_{12})}\right) = \frac{\mathbf{Y}'[(\mathbf{I}_a \otimes [\mathbf{I}_b-b^{-1}\mathbf{J}_b])-\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W']\mathbf{Y}}{a(b-1)-1}. \qquad (17)$$

Table II. Analysis of Variance Table Without Replication

Source	df	Sum of Squares
Between	a	$\mathbf{Y}'[\mathbf{I}_a \otimes b^{-1}\mathbf{J}_b]\mathbf{Y}$
Mean	1	$\mathbf{Y}'\mathbf{A}_8\mathbf{Y}$
Regression	1	$\mathbf{Y}'\mathbf{A}_9\mathbf{Y}$
Residual	a-2	$\mathbf{Y}'\mathbf{A}_{10}\mathbf{Y}$
Within	a(b-1)	$\mathbf{Y}'[\mathbf{I}_a \otimes (\mathbf{I}_b-b^{-1}\mathbf{J}_b)]\mathbf{Y}$
Regression	1	$\mathbf{Y}'\mathbf{A}_{11}\mathbf{Y}$
Residual	a(b-1)-1	$\mathbf{Y}'\mathbf{A}_{12}\mathbf{Y}$
Total	ab	$\mathbf{Y}'[\mathbf{I}_a \otimes \mathbf{I}_b]\mathbf{Y}$

Also since

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_{10}\mathbf{Y}}{tr(\mathbf{A}_{10})}\right]=E\left[\frac{\mathbf{Y}'[(\mathbf{I}_a-a^{-1}\mathbf{J}_a-\mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime}\otimes b^{-1}\mathbf{J}_b]\mathbf{Y}}{a-2}\right]$$

$$= \sigma_\varepsilon^2 + b\sigma_R^2$$

an unbiased estimator of σ_R^2 is

$$\hat{\sigma}_R^2 = b^{-1}\left[\frac{\mathbf{Y}'\mathbf{A}_{10}\mathbf{Y}}{tr(\mathbf{A}_{10})} - \frac{\mathbf{Y}'\mathbf{A}_{12}\mathbf{Y}}{tr(\mathbf{A}_{12})}\right]$$

$$= b^{-1}\mathbf{Y}'\left[\frac{(\mathbf{I}_a-a^{-1}\mathbf{J}_a-\mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime}\otimes b^{-1}\mathbf{J}_b}{a-2}\right]\mathbf{Y}$$

$$- b^{-1}\mathbf{Y}'\left[\frac{(\mathbf{I}_a\otimes[\mathbf{I}_b-b^{-1}\mathbf{J}_b])-\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W'}{a(b-1)-1}\right]\mathbf{Y}. \qquad (18)$$

<u>Simple Covariance Structure</u>

If it is assumed that $\sigma_R^2 = 0$, then the covariance structure is simplified so that $\Sigma=\sigma_\varepsilon^2(\mathbf{I}_a\otimes\mathbf{I}_r\otimes\mathbf{I}_b)$. Replication is also not necessary for this situation; hence the ANOVA table in Table II may be used to construct unbiased estimators of σ_ε^2. While the estimator of σ_ε^2 in (17) may be used, an alternative is to construct the pooled estimator

$$\hat{\sigma}_{\varepsilon p}^2 = \frac{(a-2)\mathbf{Y}'\mathbf{A}_3\mathbf{Y} + (a(b-1)-1)\mathbf{Y}'\mathbf{A}_5\mathbf{Y}}{(a-2) + (a(b-1)-1)}. \qquad (19)$$

CHAPTER V

SELECTION OF THE BEST ESTIMATOR

The practical purpose of this project is to construct tables for determining which of the least squares estimators is the best estimator to use for given parameters. The parameters which need to be considered must provide a measure of the bias, a measure of the correlation between the independent variable and the bias, and a measure of the ratio of the variance components.

The best estimator can then be chosen by calculating the relative efficiencies of the estimators, where the relative efficiencies are functions of the parameters of interest. In addition, it must be determined if the parameters in question can be estimated from the data.

Parameters

To determine appropriate parameters to measure the bias and the correlation between $\mathbf{X}_1$ and $\mathbf{X}_2$ for the balanced design, consider the expectation of the OLSE, $\hat{\beta}_1 = (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{Y}$. From model (2), $\mathbf{Y} = (\mathbf{1}_a \otimes \mathbf{1}_r \otimes \mathbf{1}_b)\beta_0 + \mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)} + \boldsymbol{\varepsilon}$. Rewriting in the form of the general linear model yields $\mathbf{Y} = \mathbf{X}\beta + \boldsymbol{\varepsilon}$ where the $px1$ vector $\beta = (\beta_0, \beta_1, \beta^{(2)\prime})'$ and the nxp matrix $\mathbf{X} = (\mathbf{1}_a \otimes \mathbf{1}_r \otimes \mathbf{1}_b \quad \mathbf{X}_1 \quad \mathbf{X}_2)$. Then

$$E(\hat{\beta}_1) = (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}\beta$$

$$E(\hat{\beta}_1) = (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'(\mathbf{1}_a\otimes\mathbf{1}_r\otimes\mathbf{1}_b \quad \mathbf{X}_1 \quad \mathbf{X}_2)(\beta_0,\ \beta_1,\ \beta^{(2)\prime})'$$

$$= (\mathbf{X}_1'\mathbf{X}_1)^{-1}(\mathbf{0} \quad \mathbf{X}_1'\mathbf{X}_1 \quad \mathbf{X}_1'\mathbf{X}_2)(\beta_0,\ \beta_1,\ \beta^{(2)\prime})'$$

$$= \beta_1 + (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}.$$

Then the bias of the OLSE is $\text{bias}(\hat{\beta}_1) = (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$ and the squared bias is

$$[\text{bias}(\hat{\beta}_1)]^2 = \frac{\beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_1\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}}{(\mathbf{X}_1'\mathbf{X}_1)^2}.$$

Since $\mathbf{X}_1'\mathbf{X}_2$ measures the correlation between $\mathbf{X}_1$ and $\mathbf{X}_2$, it is obvious that if there is no bias, $\text{bias}(\hat{\beta}_1) = 0$, then the bias and the independent variable $\mathbf{X}_1$ cannot be correlated. Also note that $\mathbf{X}_1'\mathbf{X}_2 = 0$ (no correlation) implies that the bias is equal to zero. Therefore, only one parameter is needed to measure both the bias and the correlation between $\mathbf{X}_1$ and $\mathbf{X}_2$. Let this parameter be denoted by λ.

Define $\lambda = [\text{bias}(\hat{\beta}_1)]^2/V(\hat{\beta}_1^W)$. This gives the relative size of the squared bias of the ordinary least squares estimator to the variance of the OLSWE, a useful ratio when considering the relative efficiencies of the estimators. If the bias of the OLSE is large compared to the variance of the OLSWE, then the OLSWE will be a better estimator of β_1. Note that $0 \le \lambda < \infty$.

Define $\rho = \sigma_R^2/\sigma_\varepsilon^2$ where $\rho \ge 0$. If $\rho = 0$, then $\Sigma = \sigma_\varepsilon^2(\mathbf{I}_a\otimes\mathbf{I}_r\otimes\mathbf{I}_b)$. The relative efficiencies will be developed for the weighted least squares estimator, using $\Sigma = \sigma_R^2(\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{J}_b) + \sigma_\varepsilon^2(\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b)$, then simplified for the ordinary least squares estimator, assuming $\rho = 0$.

One other parameter is necessary to make the selection of the best estimator invariant to the scale and location of the $\mathbf{X}_1$ measurements. Recall that for the balanced design with replication, $\mathbf{R}_1 = (\mathbf{I}_a \otimes \mathbf{I}_r \otimes b^{-1}\mathbf{J}_b)$ and $\mathbf{R}_2 = [\mathbf{I}_a \otimes \mathbf{I}_r \otimes (\mathbf{I}_b - b^{-1}\mathbf{J}_b)]$. Define $\gamma = \mathbf{X}_1'\mathbf{R}_2\mathbf{X}_1/\mathbf{X}_1'\mathbf{X}_1$, which measures the relative spread of the $\mathbf{X}_1$ measurements within classifications to the total spread in the $\mathbf{X}_1$ values. Then $1 - \gamma = \mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1/\mathbf{X}_1'\mathbf{X}_1$, $0 < \gamma < 1$, and $0 < 1 - \gamma < 1$.

The squared bias of the OLSBE can be written as a function the bias of the OLSE and γ. The expected value for the OLSBE is

$$E(\hat{\beta}_1^B) = (\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{R}_1\mathbf{X}\beta$$

$$= (\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{R}_1(\mathbf{1}_a \otimes \mathbf{1}_r \otimes \mathbf{1}_b \quad \mathbf{X}_1 \quad \mathbf{X}_2)(\beta_0,\ \beta_1,\ \beta^{(2)\prime})'$$

$$= (\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^{-1}\mathbf{X}_1'(\mathbf{I}_a \otimes \mathbf{I}_r \otimes b^{-1}\mathbf{J}_b)(\mathbf{1}_a \otimes \mathbf{1}_r \otimes \mathbf{1}_b \quad \mathbf{X}_1 \quad \mathbf{X}_2) \times (\beta_0,\ \beta_1,\ \beta^{(2)\prime})'$$

$$= (\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^{-1}(\mathbf{0} \quad \mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1 \quad \mathbf{X}_1'\mathbf{R}_2\mathbf{X}_2)(\beta_0,\ \beta_1,\ \beta^{(2)\prime})'$$

$$= \beta_1 + (\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}.$$

Hence, the squared bias of the OLSBE is

$$[\text{bias}(\hat{\beta}_1^B)]^2 = \frac{\beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_1\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}}{(\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^2}$$

$$[\text{bias}(\hat{\beta}_1^B)]^2 = \frac{\beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_1\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}/(\mathbf{X}_1'\mathbf{X}_1)^2}{(\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1)^2/(\mathbf{X}_1'\mathbf{X}_1)^2}$$

$$= \frac{[\text{bias}(\hat{\beta}_1)]^2}{(1 - \gamma)^2}.$$

Similarly, the squared bias of the WLSE can be written as a function of the squared bias of the OLSE and γ. The expectation of $\hat{\beta}_1^*$ is

$$E(\hat{\beta}_1^*) = (\mathbf{X}_1'\Sigma^{-1}\mathbf{X}_1)^{-1}\mathbf{X}_1'\Sigma^{-1}\mathbf{X}\beta$$

$$= (\mathbf{X}_1'\Sigma^{-1}\mathbf{X}_1)^{-1}\mathbf{X}_1'\Sigma^{-1}(\mathbf{1}_a\otimes\mathbf{1}_r\otimes\mathbf{1}_b \quad \mathbf{X}_1 \quad \mathbf{X}_2)(\beta_0,\ \beta_1,\ \beta^{(2)\prime})'$$

$$= (\mathbf{X}_1'\Sigma^{-1}\mathbf{X}_1)^{-1}\mathbf{X}_1'\Sigma^{-1}(\mathbf{I}_a\otimes\mathbf{I}_r\otimes b^{-1}\mathbf{J}_b)(\mathbf{1}_a\otimes\mathbf{1}_r\otimes\mathbf{1}_b \ \mathbf{X}_1 \ \mathbf{X}_2) \times(\beta_0,\ \beta_1,\ \beta^{(2)\prime})'$$

$$= (\mathbf{X}_1'\Sigma^{-1}\mathbf{X}_1)^{-1}(\mathbf{0} \quad \mathbf{X}_1'\Sigma^{-1}\mathbf{X}_1 \quad \mathbf{X}_1'\Sigma^{-1}\mathbf{X}_2)(\beta_0,\ \beta_1,\ \beta^{(2)\prime})'$$

$$= \beta_1 + (\mathbf{X}_1'\Sigma^{-1}\mathbf{X}_1)^{-1}\mathbf{X}_1'\Sigma^{-1}\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + (b\sigma_R^2+\sigma_\varepsilon^2)^{-1}(\mathbf{X}_1'\Sigma^{-1}\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + (b\sigma_R^2+\sigma_\varepsilon^2)^{-1}\left[\frac{\mathbf{X}_1'\mathbf{R}_1\mathbf{X}_1}{b\sigma_R^2+\sigma_\varepsilon^2} + \frac{\mathbf{X}_1'\mathbf{R}_2\mathbf{X}_1}{\sigma_\varepsilon^2}\right]^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + (b\sigma_R^2+\sigma_\varepsilon^2)^{-1}\left[\frac{\sigma_\varepsilon^2\mathbf{X}_1'\mathbf{X}_1 + b\sigma_R^2\mathbf{X}_1'\mathbf{R}_2\mathbf{X}_1}{\sigma_\varepsilon^2(b\sigma_R^2+\sigma_\varepsilon^2)}\right]^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

$$= \beta_1 + \left[\frac{\sigma_\varepsilon^2}{\sigma_\varepsilon^2\mathbf{X}_1'\mathbf{X}_1+b\sigma_R^2\mathbf{X}_1'\mathbf{R}_2\mathbf{X}_1}\right]\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}.$$

The squared bias of the WLSE is

$$[\text{bias}(\hat{\beta}_1^*)]^2 = \frac{(\sigma_\varepsilon^2)^2\beta^{(2)\prime}\mathbf{X}_2{}'\mathbf{X}_1\mathbf{X}_1{}'\mathbf{X}_2\beta^{(2)}}{(\sigma_\varepsilon^2\mathbf{X}_1{}'\mathbf{X}_1+b\sigma_R^2\mathbf{X}_1{}'\mathbf{R}_2\mathbf{X}_1)^2}$$

$$= \frac{(\sigma_\varepsilon^2)^2\beta^{(2)\prime}\mathbf{X}_2{}'\mathbf{X}_1\mathbf{X}_1{}'\mathbf{X}_2\beta^{(2)}/(\mathbf{X}_1{}'\mathbf{X}_1)^2}{(\sigma_\varepsilon^2\mathbf{X}_1{}'\mathbf{X}_1+b\sigma_R^2\mathbf{X}_1{}'\mathbf{R}_2\mathbf{X}_1)^2/(\mathbf{X}_1{}'\mathbf{X}_1)^2}$$

$$= \frac{[\text{bias}(\hat{\beta}_1)]^2}{(1 + b\rho\gamma)^2}.$$

Relative Efficiencies of the Estimators

The relative efficiencies of the estimators will be utilized to determine the best estimator for given sets of parameter values. The relative efficiency of $\hat{\beta}_1^{q1}$ to $\hat{\beta}_1^{q2}$ is defined to be

$$\text{RE}(\hat{\beta}_1^{q1}, \hat{\beta}_1^{q2}) = \frac{\text{MSE}(\hat{\beta}_1^{q1})}{\text{MSE}(\hat{\beta}_1^{q2})} = \frac{V(\hat{\beta}_1^{q1}) + [\text{bias}(\hat{\beta}_1^{q1})]^2}{V(\hat{\beta}_1^{q2}) + [\text{bias}(\hat{\beta}_1^{q2})]^2}.$$

Since $\Sigma = \sigma_R^2(\mathbf{I}_a\otimes\mathbf{I}_r\otimes\mathbf{J}_b) + \sigma_\varepsilon^2(\mathbf{I}_a\otimes\mathbf{I}_r\otimes\mathbf{I}_b)$, the variances of the estimators are $V(\hat{\beta}_1^*) = \sigma_\varepsilon^2(b\sigma_R^2+\sigma_\varepsilon^2)(\sigma_\varepsilon^2\mathbf{X}_1{}'\mathbf{X}_1+b\sigma_R^2\mathbf{X}_1{}'\mathbf{R}_2\mathbf{X}_1)^{-1}$, $V(\hat{\beta}_1^B) = (b\sigma_R^2+\sigma_\varepsilon^2)(\mathbf{X}_1{}'\mathbf{R}_1\mathbf{X}_1)^{-1}$ and $V(\hat{\beta}_1^W) = \sigma_\varepsilon^2(\mathbf{X}_1{}'\mathbf{R}_2\mathbf{X}_1)^{-1}$. Then the relative efficiency of $\hat{\beta}_1^*$ to $\hat{\beta}_1^B$ is

$$\text{RE}(\hat{\beta}_1^*, \hat{\beta}_1^B) = \frac{\text{MSE}(\hat{\beta}_1^*)}{\text{MSE}(\hat{\beta}_1^B)} = \frac{V(\hat{\beta}_1^*) + [\text{bias}(\hat{\beta}_1^*)]^2}{V(\hat{\beta}_1^B) + [\text{bias}(\hat{\beta}_1^B)]^2}$$

$$= \frac{(V(\hat{\beta}_1^*) + [\text{bias}(\hat{\beta}_1^*)]^2)/V(\hat{\beta}_1^W)}{(V(\hat{\beta}_1^B) + [\text{bias}(\hat{\beta}_1^B)]^2)/V(\hat{\beta}_1^W)}$$

$$RE(\hat{\beta}_1^*,\ \hat{\beta}_1^B) = \frac{V(\hat{\beta}_1^*)/V(\hat{\beta}_1^W) + [bias(\hat{\beta}_1)]^2/(1+b\rho\gamma)^2 V(\hat{\beta}_1^W)}{V(\hat{\beta}_1^B)/V(\hat{\beta}_1^W) + [bias(\hat{\beta}_1)]^2)/(1-\gamma)^2 V(\hat{\beta}_1^W)}$$

$$= \frac{[(1+b\rho)\gamma]/(1+b\rho\gamma) + \lambda/(1+b\rho\gamma)^2}{[(1+b\rho)\gamma]/(1-\gamma) + \lambda/(1-\gamma)^2}$$

$$= \frac{(1-\gamma)^2[(1+b\rho\gamma)(1+b\rho)\gamma + \lambda]}{(1+b\rho\gamma)^2[(1-\gamma)(1+b\rho)\gamma + \lambda]}.$$

Since $(1-\gamma) < (1+bp\gamma)$, then $(1-\gamma)(1+b\rho\gamma)(1+b\rho)\gamma+(1-\gamma)\lambda <$ $(1+b\rho\gamma)(1-\gamma)(1+b\rho)\gamma+(1+b\rho\gamma)\lambda$. Hence,

$$(1-\gamma)[(1-\gamma)(1+b\rho\gamma)(1+b\rho)\gamma+(1-\gamma)\lambda] < (1+b\rho\gamma)[(1+b\rho\gamma)(1-\gamma)(1+b\rho)\gamma+(1+b\rho\gamma)\lambda].$$

Hence, $RE(\hat{\beta}_1^*,\ \hat{\beta}_1^B) < 1$ for all $0 \le \lambda < \infty$, $\rho \ge 0$, $0 < \gamma < 1$, and $b \ge 2$. This implies that the WLSE is always a better estimator for β_1 than the OLSBE.

If $\rho = 0$, then $\Sigma = \sigma_\varepsilon^2(\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b)$, and the relative efficiency, $RE(\hat{\beta}_1^*,\ \hat{\beta}_1^B)$, simplifies to

$$RE(\hat{\beta}_1,\ \hat{\beta}_1^B) = \frac{(1-\gamma)^2(\gamma+\lambda)}{(1-\gamma)\gamma + \lambda}.$$

Since $1-\gamma < 1$, $(1-\gamma)^2\gamma < (1-\gamma)\gamma$, so $(1-\gamma)^2\gamma + (1-\gamma)^2\lambda <$ $(1-\gamma)\gamma + \lambda$. Therefore, $RE(\hat{\beta}_1,\ \hat{\beta}_1^B) < 1$. Then the OLSE is always a better estimator than the OLSBE.

Since the OLSBE is never more efficient than the WLSE, the experimenter need only compare the OLSWE with the WLSE to determine the best estimator. The relative efficiency of the WLSE to the OLSWE is

$$\mathrm{RE}(\hat{\beta}_1^*,\ \hat{\beta}_1^W) = \frac{\mathrm{MSE}(\hat{\beta}_1^*)}{\mathrm{MSE}(\hat{\beta}_1^W)} = \frac{V(\hat{\beta}_1^*) + [\mathrm{bias}(\hat{\beta}_1^*)]^2}{V(\hat{\beta}_1^W)}$$

$$= \frac{V(\hat{\beta}_1^*)}{V(\hat{\beta}_1^W)} + \frac{[\mathrm{bias}(\hat{\beta}_1^*)]^2}{V(\hat{\beta}_1^W)}$$

$$= \frac{(1+b\rho)\gamma}{(1+b\rho\gamma)} + \frac{[\mathrm{bias}(\hat{\beta}_1)]^2}{V(\hat{\beta}_1^W)(1+b\rho\gamma)^2}$$

$$= \frac{(1+b\rho\gamma)(1+b\rho)\gamma + \lambda}{(1+b\rho\gamma)^2}.$$

Note that when $\lambda = 0$, i.e. there is no significant bias, this relative efficiency reduces to $\mathrm{RE}(\hat{\beta}_1^*,\ \hat{\beta}_1^W) = (1+b\rho)\gamma/(1+b\rho\gamma)$. Then $\mathrm{RE}(\hat{\beta}_1^*,\ \hat{\beta}_1^W) < 1$ since $\gamma < 1$. This indicates that the WLSE should be used when there is no bias.

If $\lambda \neq 0$, then the OLSWE is a more efficient estimator of β_1 than the WLSE when $\lambda > (1+b\rho\gamma)(1-\gamma)$. Thus, when bias is large relative to ρ, $\hat{\beta}_1^W$ is a better estimator than $\hat{\beta}_1^*$. Since the WLSE accounts for the covariance structure, as ρ becomes large relative to λ and as b increases, the WLSE becomes a better estimator for β_1.

When $\rho = 0$, the relative efficiency of the OLSE to the OLSWE is $\mathrm{RE}(\hat{\beta}_1,\ \hat{\beta}_1^W) = \gamma + \lambda$. Then $\hat{\beta}_1^W$ is more efficient than $\hat{\beta}_1$ when $\lambda > 1-\gamma$.

The relative efficiency values for $\mathrm{RE}(\hat{\beta}_1^*,\ \hat{\beta}_1^W)$ for the following combinations of parameter values may be found in Appendix B, along with the SAS program used to generate the values: λ = 0, 0.5, 1, 2, 5; γ = 0.001, 0.2, 0.5, 0.7, 0.990; ρ = 0, 0.5, 1, 2, 5; and b = 2, 3, 6, 12. Appendix C provides the program and output for the special case when ρ = 0 where the OLSE is compared to the OLSWE, and Appendix D gives the program and output for the special case when there is no significant bias, λ = 0.

Table III, on the following page, specifies which estimator is most efficient for the given sets of parameter values. Of particular interest is the combination of λ = 1 and ρ = 0.5. As b increases, the WLSE becomes the better estimator of β_1 for more values of γ. When b = 2, the WLSE is used only when γ = 0.001. However, the WLSE is most efficient for γ values of 0.001 and 0.2 for b = 3, γ values of 0.001, 0.2, and 0.5 for b = 6, and γ values of 0.001, 0.2, 0.5 and 0.7 when b = 12. Similarly, the combinations of λ = 2 for ρ values of 2 and 5, and λ = 5 for ρ values of 2 and 5 specify different best estimators for certain γ values as the number of levels of time increase. For γ = 0.990, λ = 0.5 (small bias), ρ = 0 (large variance ratio), the WLSE is most efficient for b = 12.

Note that if a design without replications is utilized, the definitions of the parameters remain the same. The $\mathbf{Y}$, $\mathbf{X}_1$, $\mathbf{R}_1$, $\mathbf{R}_2$ and Σ matrices are simplified so that $\mathbf{Y} = (y_{11}, \ldots, y_{1b}, \ldots, y_{a1}, \ldots, y_{ab})'$, $\mathbf{X}_1 = (x_{11}-\bar{x}_{..}, \ldots, x_{1b}-\bar{x}_{..}, \ldots, x_{a1}-\bar{x}_{..}, \ldots, x_{ab}-\bar{x}_{..})'$, $\mathbf{R}_1 = (\mathbf{I}_a \otimes b^{-1}\mathbf{J}_b)$, $\mathbf{R}_2 = \mathbf{I}_a \otimes (\mathbf{I}_b - b^{-1}\mathbf{J}_b)$ and $\Sigma = \sigma_R^2(\mathbf{I}_a \otimes \mathbf{J}_b) + \sigma_\varepsilon^2(\mathbf{I}_a \otimes \mathbf{I}_b)$. Hence Table III may also be used for the design with no replications.

Table III. Most Efficient Estimator for Specified Combinations of λ, γ, ρ, and b

		ρ											
		0				0.5				1			
	b	2	3	6	12	2	3	6	12	2	3	6	12
λ	γ												
0.0	.001	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.200	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.500	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.700	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.990	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
0.5	.001	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.200	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.500	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.700	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.990	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
1.0	.001	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.200	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.500	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.700	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.990	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
2.0	.001	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
	.200	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$
	.500	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$
	.700	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$
	.990	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
5.0	.001	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
	.200	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
	.500	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
	.700	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
	.990	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$

continued on following page

Table III continued. Most Efficient Estimator for Specified Combinations of λ, γ, ρ, and b

		ρ = 2				ρ = 5			
	b	2	3	6	12	2	3	6	12
λ	γ								
0.0	.001	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.200	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.500	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.700	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.990	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
0.5	.001	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.200	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.500	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.700	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.990	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$
1.0	.001	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.200	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.500	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.700	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.990	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
2.0	.001	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
	.200	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.500	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.700	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$	$\hat{\beta}_1^*$
	.990	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
5.0	.001	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$
	.200	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$
	.500	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$
	.700	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^*$
	.990	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$	$\hat{\beta}_1^W$

Estimating the Bias Term

It was determined in Chapter IV that σ_R^2 and σ_ε^2 may be estimated from the sample, hence an estimator of ρ may be calculated. The parameter γ is a function of the $\mathbf{X}_1$ matrix which is known, hence γ is known. Since the expected mean square of the between classifications residual lack of fit portion is $\beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_2\beta^{(2)} + \beta^{(2)\prime}\mathbf{X}_2'[\mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime} \otimes b^{-1}\mathbf{J}_b]\mathbf{X}_2\beta^{(2)}$, it is not possible to estimate the bias from the ANOVA table. To use Table III, one would have to either assume the bias term is known, or there exists an estimate of the bias from a source outside the data.

One might use information from the ANOVA table to estimate λ if the researcher had an outside estimate of the R^2 value that was anticipated prior to taking data, where R^2 is the ratio of the sum of squares for regression to the total sum of squares. This outside estimate of R^2 might be found in previous work in the field of interest.

If the R^2 calculated from the data is much smaller than anticipated, one might suspect that more explanatory variables are necessary and the $\mathbf{X}_2\beta^{(2)}$ term should be included in the model. An estimator of λ may be constructed using the anticipated and the calculated values.

Let $R_{1,2}^2$ denote the anticipated R^2 value which would assume model (2) is correct, and $SSR_{1,2}$ be the sum of squares for regression from that model, which would be calculated using both $\mathbf{X}_1$ and $\mathbf{X}_2$. Then $R_{1,2}^2 = SSR_{1,2}/SST$

where SST is the total sum of squares. Let $\mathbf{X}_{B2} = (\mathbf{X}_B\ \mathbf{X}_2)$, $\mathbf{X}_{W2} = (\mathbf{X}_W\ \mathbf{0})$, both of dimension *arb*x*(p-1)*, and let $\mathbf{X} = (\mathbf{X}_1\ \mathbf{X}_2)$. Then Table IV illustrates the ANOVA table when model (2) is assumed to be the correct model.

Adding the sum of squares for between and within regression,

$$R^2_{1,2} = \frac{SSR_{1,2}}{SST}$$

$$= \frac{\mathbf{Y}'\,[\mathbf{X}_{B2}(\mathbf{X}_{B2}'\mathbf{X}_{B2})^{-1}\mathbf{X}_{B2}' + \mathbf{X}_{W2}(\mathbf{X}_{W2}'\mathbf{X}_{W2})^{-1}\mathbf{X}_{W2}']\,\mathbf{Y}}{\mathbf{Y}'\,[\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b]\,\mathbf{Y}}$$

$$= \frac{\mathbf{Y}'\,[\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}']\,\mathbf{Y}}{\mathbf{Y}'\,[\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b]\,\mathbf{Y}}.$$

Table IV. Analysis of Variance Table Assuming Model (2)

Source	df	Sum of Squares
Between	ar	$\mathbf{Y}'\,[\mathbf{I}_a \otimes \mathbf{I}_r \otimes b^{-1}\mathbf{J}_b]\,\mathbf{Y}$
Mean	1	$\mathbf{Y}'\,[a^{-1}\mathbf{J}_a \otimes r^{-1}\mathbf{J}_r \otimes b^{-1}\mathbf{J}_b]\,\mathbf{Y}$
Regression	1	$\mathbf{Y}'\,[\mathbf{X}_{B2}(\mathbf{X}_{B2}'\mathbf{X}_{B2})^{-1}\mathbf{X}_{B2}']\,\mathbf{Y}$
Residual	ar-2	$\mathbf{Y}'\,[\{(\mathbf{I}_a \otimes \mathbf{I}_r) - (a^{-1}\mathbf{J}_a \otimes r^{-1}\mathbf{J}_r) \otimes b^{-1}\mathbf{J}_b\} - \mathbf{X}_{B2}(\mathbf{X}_{B2}'\mathbf{X}_{B2})^{-1}\mathbf{X}_{B2}']\,\mathbf{Y}$
Within	ar(b-1)	$\mathbf{Y}'\,[\mathbf{I}_a \otimes \mathbf{I}_r \otimes (\mathbf{I}_b - b^{-1}\mathbf{J}_b)]\,\mathbf{Y}$
Regression	1	$\mathbf{Y}'\,[\mathbf{X}_{W2}(\mathbf{X}_{W2}'\mathbf{X}_{W2})^{-1}\mathbf{X}_{W2}']\,\mathbf{Y}$
Residual	ar(b-1)-1	$\mathbf{Y}'\,[(\mathbf{I}_a \otimes \mathbf{I}_r \otimes [\mathbf{I}_b - b^{-1}\mathbf{J}_b]) - \mathbf{X}_{W2}(\mathbf{X}_{W2}'\mathbf{X}_{W2})^{-1}\mathbf{X}_{W2}']\,\mathbf{Y}$
Total	ab	$\mathbf{Y}'\,[\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b]\,\mathbf{Y}$

Let R_1^2 be the R^2 value calculated from the data, and SSR_1 be the sum of squares for regression utilizing only the $\mathbf{X}_1$ matrix, assuming the model $\mathbf{Y} = \mathbf{1}_n\beta_0 + \mathbf{X}_1\beta_1 + \varepsilon$. Then from Table I,

$$R_1^2 = \frac{SSR_1}{SST}$$

$$= \frac{\mathbf{Y}[\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B']\mathbf{Y}}{\mathbf{Y}'[\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b]\mathbf{Y}}$$

$$= \frac{\mathbf{Y}[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1']\mathbf{Y}}{\mathbf{Y}'[\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b]\mathbf{Y}}.$$

Consider the quantity

$$R_{1,2}^2 - R_1^2 = \frac{SSR_{1,2}}{SST} - \frac{SSR_1}{SST}.$$

This is a measure of the difference between the anticipated and the calculated values for R^2. The parameter λ can be expressed as a function of the expectation of this quantity, the expected lack of fit sum of squares (SSLOF), the $\mathbf{X}_1$ matrix and γ. Now

$$E(SSR_{1,2}) = \beta_1'\mathbf{X}_1'\mathbf{X}_1\beta_1 + \beta_1'\mathbf{X}_1'\mathbf{X}_2\beta_2 + \beta_2'\mathbf{X}_2'\mathbf{X}_1\beta_1 + \beta_2'\mathbf{X}_2'\mathbf{X}_2\beta_2,$$

$$E(SSR_1) = \beta_1'\mathbf{X}_1'\mathbf{X}_1\beta_1 + \beta_1'\mathbf{X}_1'\mathbf{X}_2\beta_2 + \beta_2'\mathbf{X}_2'\mathbf{X}_1\beta_1 + \beta_2'\mathbf{X}_2'\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta_2,$$

and

$$E(SSLOF) = \beta_2'\mathbf{X}_2'\mathbf{X}_2\beta_2 + \beta_2'\mathbf{X}_2'\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B'\mathbf{X}_2\beta_2.$$

Then

$$E(SSR_{1,2}) - E(SSR_1) - E(SSLOF) = \beta_1'\mathbf{X}_1'\mathbf{X}_1\beta_1 + \beta_1'\mathbf{X}_1'\mathbf{X}_2\beta_2$$
$$+ \beta_2'\mathbf{X}_2'\mathbf{X}_1\beta_1 + \beta_2'\mathbf{X}_2'\mathbf{X}_2\beta_2 - \beta_1'\mathbf{X}_1'\mathbf{X}_1\beta_1 - \beta_1'\mathbf{X}_1'\mathbf{X}_2\beta_2$$
$$- \beta_2'\mathbf{X}_2'\mathbf{X}_1\beta_1 - \beta_2'\mathbf{X}_2'\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta_2 - \beta_2'\mathbf{X}_2'\mathbf{X}_2\beta_2$$
$$+ \beta_2'\mathbf{X}_2'\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B'\mathbf{X}_2\beta_2$$

$$= \beta_2'\mathbf{X}_2'\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B'\mathbf{X}_2\beta_2 - \beta_2'\mathbf{X}_2'\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta_2$$

$$= (\mathbf{X}_B'\mathbf{X}_B)^{-1}(\mathbf{X}_1'\mathbf{X}_1)^2\lambda - (\mathbf{X}_1'\mathbf{X}_1)\lambda$$

$$= (\mathbf{X}_1'\mathbf{X}_1)\gamma(1-\gamma)^{-1}\lambda.$$

Since $SST(R^2_{1,2})$ is an estimator for $SSR_{1,2}$ and estimators of SSR_1 and SSLOF are $\mathbf{Y}[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1']\mathbf{Y}$ and $\mathbf{Y}'[(\{(\mathbf{I}_a - a^{-1}\mathbf{J}_a)\otimes r^{-1}\mathbf{J}_r\} - \mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime} - \bigoplus_{i=1}^{a}\mathbf{B}_i(\mathbf{B}_i'\mathbf{B}_i)^{-1}\mathbf{B}_i')\otimes b^{-1}\mathbf{J}_b]\mathbf{Y}$, respectively, then an estimator of λ for $R^2_{1,2}$ known is

$$\hat{\lambda} = (1-\gamma)[(\mathbf{X}_1'\mathbf{X}_1)\gamma]^{-1}[SST(R^2_{1,2}) - \mathbf{Y}[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1']\mathbf{Y}$$
$$- \mathbf{Y}'[(\{(\mathbf{I}_a - a^{-1}\mathbf{J}_a)\otimes r^{-1}\mathbf{J}_r\} - \mathbf{X}_B^*(\mathbf{X}_B^{*\prime}\mathbf{X}_B^*)^{-1}\mathbf{X}_B^{*\prime}$$
$$- \bigoplus_{i=1}^{a}\mathbf{B}_i(\mathbf{B}_i'\mathbf{B}_i)^{-1}\mathbf{B}_i')\otimes b^{-1}\mathbf{J}_b]\mathbf{Y}].$$

Recall that replications are necessary for the estimator of SSLOF (from Table I).

CHAPTER VI

SIMULATIONS

Simulations were run to determine how often the most efficient estimator would be selected when used in practice. Simulated samples were produced utilizing one of three FORTRAN programs, all of which may be found in Appendix E.

Values Generated and Calculated

The elements of the $\mathbf{X}_1$ vector were generated to produce the requested values for γ from a Uniform(0,1) distribution. For the purpose of the simulations, it was assumed that an estimator of λ was known. The first element of $\mathbf{X}_2$ was calculated from λ, assuming that the second element was the negative of the first and the third element was 0 for a = 3. When a = 5, the first element was calculated from λ, assuming the second element was the negative of the first, the third element was the the first, the fourth element was half the second and the last element was 0.

The elements of the $\mathbf{X}_1$ and $\mathbf{X}_2$ vectors were standardized so that the scale of measurement would not affect the selection of the estimator for β_1. Since none of the parameters are functions of β_0 or β_1, nor is the relative efficiency of the WLSE to the OLSWE, values of $\beta_0 = 0$ and $\beta_1 = 1$, were selected for these parameters.

One thousand samples were created for each of the following combinations of parameter values: a = 3 and 5; b = 2, 3, 6 and 12; γ = 0.001, 0.2, 0.5, 0.7 and 0.99; λ = 0, 0.5, 1, 2 and 5; and ρ = 0, 0.5, 1, 2 and 5. **Y** vectors were generated using the model $\mathbf{Y} = \mathbf{X}_1 + \mathbf{X}_2\beta^{(2)} + \varepsilon$. The random normal number generator was used to generate two error terms, one associated with the between portion of the model and one associated with the within.

An estimator of ρ was calculated using the estimators of σ_R^2 and σ_ε^2 as indicated in Chapter IV. When a negative estimator was calculated, $\hat{\sigma}_R^2$ was set to zero. Pooled estimators of σ_ε^2 were not used in the simulations.

The relative efficiency of the weighted least squares estimator to the ordinary least squares within estimator was calculated for each sample and the most efficient estimator selected. This selection was then compared to the estimator indicated as the most efficient for the given parameter values in Table III. The frequency of correct selection was recorded and the percentage was calculated, utilizing a SAS program which may be found in Appendix F. Frequency tables for all combinations of λ and ρ were produced for each a, b and γ combination.

Results of Simulations

The results of the simulations indicate that the correct estimator is selected 89% of the time, summed over all combinations of the parameters considered, when the variance ratio is estimated from the data. The number of correct decisions in each cell (λ and ρ combination) for each frequency table, as well as the percentage, may be found in Appendix G.

Because the relative efficiency is the ratio of the mean squares of the WLSE and the OLSWE, the choice of the estimator as shown in Table III is based on the value of the relative efficiency as compared to one. If the relative efficiency is greater than one, the OLSWE is selected; if the relative efficiency is less than or equal to one, the WLSE is selected. This selection process will be referred to as Criteria 1.

Similarly, the value of the relative efficiency from each sample was compared to one, and the WLSE or OLSWE was selected based on the relationship of that value to one. The correctness of the decision was then determined by comparing the estimator selected by the sample to the estimator selected by Table III.

Since the value of one is the point at which the decision changes, it seems reasonable to assume that for relative efficiencies close to one, as calculated from the true parameter values, the selection of the best estimator is not as crucial as it would be for values much smaller or much larger than one. The best estimator in these cases is not sufficiently more efficient than the other. Thus, the selection of the appropriate estimator a vital issue. Under this assumption, the choice of the estimator within cells in which the relative efficiency is close to one may be considered to be correct without regard to whether the WLSE or OLSWE is selected.

Values for the relative efficiency between .5 and 2 were considered to be close to one. In this case, the WLSE is the correct choice if the relative efficiency is less than 0.5 and the OLSWE is the correct choice if the relative efficiency is greater than 2. Either estimator is considered to be a correct choice if the relative efficiency is between 0.5 and 2 inclusive. This selection process is referred to as Criteria 2. Table V on the following page summarizes the percentage of correct choices for each a, b and γ combination summed over values of λ and ρ for Criteria 1 and Criteria 2.

The lowest percentage of correct decisions for Criteria 1, 75.944%, occurred for the frequency table for the parameter combination a = 3, b = 2, γ = 0.7, while the highest percentage, 99.504%, occurred for the parameter combination a = 5, b = 3, γ = 0.99. For Criteria 2, the lowest percentage, 97.8%, occurred in the frequency table for the parameter combination a = 3, b = 2, γ = .2, and the highest percentage, 100%, occurred for several parameter combinations. The percentage of correct decisions tended to increase as the levels of time increased. Also, the percentage of correct decisions tended to be higher for five levels of classification than for three.

Table V. Percentage of Correct Decisions for Criteria 1 and Criteria 2 for A, B, γ Combinations Summed Over λ and ρ

A	B	γ	% Correct Criteria 1	% Correct Criteria 2
3	2	.001	91.800	99.976
		.2	83.636	97.800
		.5	82.284	98.128
		.7	75.944	98.836
		.99	96.760	99.920
3	3	.001	92.072	100.000
		.2	85.104	99.392
		.5	83.292	99.332
		.7	78.648	99.560
		.99	98.752	99.996
3	6	.001	93.612	100.000
		.2	83.772	99.760
		.5	81.516	99.796
		.7	76.916	99.880
		.99	98.116	100.000
3	12	.001	94.664	100.000
		.2	84.060	99.928
		.5	81.136	99.888
		.7	76.488	99.956
		.99	95.008	100.000
5	2	.001	94.016	100.000
		.2	88.316	99.316
		.5	86.616	99.152
		.7	82.420	99.820
		.99	99.176	100.000
5	3	.001	95.156	100.000
		.2	90.092	99.880
		.5	87.704	99.936
		.7	84.856	99.964
		.99	99.504	100.000
5	6	.001	97.012	100.000
		.2	89.908	99.980
		.5	88.600	99.988
		.7	85.312	99.996
		.99	98.920	100.000
5	12	.001	98.228	100.000
		.2	90.920	100.000
		.5	89.220	100.000
		.7	87.204	100.000
		.99	96.324	100.000

The selection of the most efficient estimator of β_1 becomes more imperative for parameter combinations for which the relative efficiency is less than 0.5 or greater than 2. The percentage of correct decisions was inspected for these cells in each of the frequency tables, and those producing the correct decision less than 90% of the time were examined.

There were no cells for which the relative efficiency was less than 0.5 where the correct estimator was selected less than 100% of the time. When the WLSE was at least twice as efficient as the OLSWE, the correct estimator, WLSE, was always selected.

The correct estimator was selected less than 90% of time in 6 cells for which the relative efficiency was greater than 2. Since the true relative efficiency is greater than one, when an incorrect decision is made, the WLSE is selected when the OLSWE is the more efficient estimator. This indicates that the selection process using Criteria 2 is conservative. If an error was consistently made in a cell, the error was always in selecting the WLSE, the estimator that would have ordinarily have been used.

The parameter combinations for the 6 cells in which the WLSE was incorrectly selected more than 10% of the time are listed in Table VI on the following page. The number of incorrect decisions and the magnitude of the maximum difference between the sample relative efficiency and the value one is given in the table.

Table VI. Parameter Combinations in Which WLSE was Incorrectly Selected More than 10% of the Time

A	B	γ	λ	ρ	Frequency	Maximum Distance
3	2	.2	2	0	163	.08
3	2	.2	5	2	160	.03
3	2	.5	2	0	164	.03
3	2	.5	5	.5	101	.01
3	2	.5	5	1	148	.01
3	2	.7	2	0	119	.01

Note that in every case, this maximum distance is less than or equal to 0.08. Therefore, even when an incorrect decision is made, the sample relative efficiency is close to the value one, indicating the selection of the correct estimator is missed by a small distance.

This error in selection appears to be due to the method of generating the $\mathbf{X}_2$ value. For some parameter combinations, particularly for small samples, large $\hat{\sigma}_R^2$ are generated. For example, for the parameter combination a = 3, b = 2, γ = 0.001, 559 estimates of ρ where generated that were greater than 100, with the maximum value being 30,517.

CHAPTER VII

SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

Least squares estimators of the slope parameter, β_1, are frequently used when the model is assumed to be known, and in fact, are the best linear unbiased estimators under certain conditions. Unfortunately, if an unknown bias term is present in the model and bias is not constant across classifications, the weighted least squares and ordinary least squares estimators become biased for β_1. Two new estimators which were derived from the weighted least squares estimator, the ordinary least squares between estimator and the ordinary least squares within estimator, were offered for consideration.

The OLSBE also is biased for β_1 when an unknown bias term is present in the model and is not constant across classifications. However, the OLSWE remains unbiased. If the variance of the OLSWE is small relative to the size of the bias in the WLSE, the OLSWE could be a viable alternative to the WLSE.

In comparing mean squares of the WLSE to the OLSWE, the relative efficiency calculations showed that the OLSWE is a more efficient estimator of β_1 for certain sets of parameter values. When the bias is large compared to size of the variance ratio, ρ, the OLSWE is often more efficient. This is particularly true for smaller samples. As ρ gets larger and the number of levels of time get larger, the WLSE becomes more efficient.

Results of simulations indicate that when estimating ρ from the data and calculating the relative efficiency of the WLSE and OLSWE, the correct estimator is selected a 84% of the time. It was also noted that for parameter combinations in which selection of the correct estimator is more vital, most errors tend to occur in a conservative manner, i.e. the WLSE is selected when the OLSWE is more efficient.

Table III allows an experimenter to determine the best estimator for the given parameter values, provided that an estimate of λ is available. Information from outside the data is necessary to produce an estimator of the bias.

Recommendations for Further Study

Since γ and b are known values from the data and $\hat{\rho}$ can be calculated using the ANOVA estimator, it would certainly enhance the practical utility of Table III in selecting the best estimator if a estimator of the bias could be produced directly from the sample. More study in generating such a statistic is indicated.

The possibility of an alternative method of generating $\mathbf{X}_2$ values should be explored. A procedure is needed that produces $\mathbf{X}_2$ values that combine with the $\mathbf{X}_1$ values for the γ and λ values needed without creating outliers that inflate the variance estimators. In addition, further simulations should be extended to determine differences in accuracy of selection when $p > 2$ so that $\mathbf{X}_2$ is a matrix with more than one column.

The behavior of the estimators should also be investigated in a multivariate setting in which the $\mathbf{X}_1$ vector becomes an *nxq* matrix. Several independent variables would then be measured over time along with the dependent variable.

BIBLIOGRAPHY

Draper, N. and Smith, H. (1981), ***Applied Regression Analysis***, New York: John Wiley and Sons.

Kempthorne, O. (1983), ***The Design and Analysis of Experiments***, Malabar, FL: Robert E. Krieger Publishing Co., Inc.

Moser, B.K. (in press), ***Linear Models: The Mean Model Approach***, San Diego: Academic Press, Inc.

Neter, J., Wasserman, W., and Kutner, M.H. (1989), ***Applied Linear Regression Models***, Homewood, IL: Irwin.

Searle, S.R. (1982), ***Matrix Algebra Useful for Statistics***, New York: John Wiley and Sons.

APPENDICES

APPENDIX A
CALCULATIONS OF EXPECTED MEAN SQUARES

Expected Mean Squares for Table I

Between Classifications

Mean

$$E\left(\frac{\mathbf{Y}'\mathbf{A}_1\mathbf{Y}}{tr(\mathbf{A}_1)}\right) = \frac{tr(\mathbf{A}_1\Sigma) + \mu'\mathbf{A}_1\mu}{tr(\mathbf{A}_1)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + (\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(a^{-1}\mathbf{J}_a \otimes r^{-1}\mathbf{J}_r \otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

Regression

$$E\left(\frac{\mathbf{Y}'\mathbf{A}_2\mathbf{Y}}{tr(\mathbf{A}_2)}\right) = \frac{tr(\mathbf{A}_2\Sigma) + \mu'\mathbf{A}_2\mu}{tr(\mathbf{A}_2)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + (\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B')(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \beta_1'\mathbf{X}_B'\mathbf{X}_B\beta_1 + \beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)} + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B\beta_1 + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_2\beta^{(2)}$$

Residual

$$E\left(\frac{\mathbf{Y}'\mathbf{A}_3\mathbf{Y}}{tr(\mathbf{A}_3)}\right) = \frac{tr(\mathbf{A}_3\Sigma) + \mu'\mathbf{A}_3\mu}{tr(\mathbf{A}_3)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \frac{(\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{I}_a \otimes \mathbf{I}_r \otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})}{ar - 2}$$
$$- \frac{(\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(a^{-1}\mathbf{J}_a \otimes r^{-1}\mathbf{J}_r \otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})}{ar - 2}$$
$$- \frac{(\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B')(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})}{ar - 2}$$

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_3\mathbf{Y}}{tr(\mathbf{A}_3)}\right] = b\sigma_R^2+\sigma_\varepsilon^2 + \frac{\beta_1'\mathbf{X}_B'\mathbf{X}_B\beta_1 - \beta_1'\mathbf{X}_B'\mathbf{X}_B\beta_1 + \beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{ar - 2}$$

$$- \frac{\beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)} - \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B\beta_1 + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B\beta_1}{ar - 2}$$

$$+ \frac{\beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_2\beta^{(2)} - \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{ar - 2}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \frac{\beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_2\beta^{(2)} - \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{ar - 2}$$

<u>Lack of Fit</u>

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_4\mathbf{Y}}{tr(\mathbf{A}_4)}\right] = \frac{tr(\mathbf{A}_4\Sigma) + \mu'\mathbf{A}_4\mu}{tr(\mathbf{A}_4)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

$$+ \frac{(\beta_1'\mathbf{X}_B'+\beta^{(2)}{}'\mathbf{X}_2')(\mathbf{I}_a\otimes r^{-1}\mathbf{J}_r\otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1+\mathbf{X}_2\beta^{(2)})}{a - 2}$$

$$- \frac{(\beta_1'\mathbf{X}_B'+\beta^{(2)}{}'\mathbf{X}_2')(a^{-1}\mathbf{J}_a\otimes r^{-1}\mathbf{J}_r\otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1+\mathbf{X}_2\beta^{(2)})}{a - 2}$$

$$- \frac{(\beta_1'\mathbf{X}_B'+\beta^{(2)}{}'\mathbf{X}_2')(\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B')(\mathbf{X}_B\beta_1+\mathbf{X}_2\beta^{(2)})}{a - 2}$$

$$- \frac{(\beta_1'\mathbf{X}_B'+\beta^{(2)}{}'\mathbf{X}_2')(\bigoplus_{i=1}^{a}\mathbf{B}_i(\mathbf{B}_i'\mathbf{B}_i)^{-1}\mathbf{B}_i)(\mathbf{X}_B\beta_1+\mathbf{X}_2\beta^{(2)})}{a - 2}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \frac{\beta_1'\mathbf{X}_B'(\mathbf{I}_a\otimes r^{-1}\mathbf{J}_r\otimes b^{-1}\mathbf{J}_b)\mathbf{X}_B\beta_1 + \beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{a - 2}$$

$$+ \frac{\beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B\beta_1 + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_2\beta^{(2)} - \beta_1'\mathbf{X}_B'(\mathbf{I}_a\otimes\mathbf{I}_r\otimes b^{-1}\mathbf{J}_b)\mathbf{X}_B\beta_1}{a - 2}$$

$$- \frac{\beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}+\beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B\beta_1+\beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{a - 2}$$

$$- \frac{\beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B(\mathbf{I}_a\otimes r^{-1}\mathbf{J}_r\otimes b^{-1}\mathbf{J}_b)\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{a - 2}$$

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_4\mathbf{Y}}{tr(\mathbf{A}_4)}\right] = b\sigma_R^2 + \sigma_\varepsilon^2$$

$$+ \frac{\beta^{(2)\prime}\mathbf{X}_2{}'\mathbf{X}_2\beta^{(2)} + \beta^{(2)\prime}\mathbf{X}_2{}'\mathbf{X}_B(\mathbf{X}_B{}'\mathbf{X}_B)^{-1}\mathbf{X}_B{}'\mathbf{X}_2\beta^{(2)}}{a - 2}$$

Pure Error

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_5\mathbf{Y}}{tr(\mathbf{A}_5)}\right] = \frac{tr(\mathbf{A}_5\Sigma) + \mu'\mathbf{A}_5\mu}{tr(\mathbf{A}_5)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

$$+ \frac{(\beta_1{}'\mathbf{X}_B{}'+\beta^{(2)\prime}\mathbf{X}_2{}')(\mathbf{I}_a\otimes\mathbf{I}_r\otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1+\mathbf{X}_2\beta^{(2)})}{a(r - 1)}$$

$$- \frac{(\beta_1{}'\mathbf{X}_B{}'+\beta^{(2)\prime}\mathbf{X}_2{}')(\mathbf{I}_a\otimes r^{-1}\mathbf{J}_r\otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1+\mathbf{X}_2\beta^{(2)})}{a(r - 1)}$$

$$- \frac{(\beta_1{}'\mathbf{X}_B{}'+\beta^{(2)\prime}\mathbf{X}_2{}')(\bigoplus_{i=1}^{a}\mathbf{B}_i(\mathbf{B}_i{}'\mathbf{B}_i)^{-1}\mathbf{B}_i)(\mathbf{X}_B\beta_1+\mathbf{X}_2\beta^{(2)})}{a(r - 1)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \frac{\beta_1{}'\mathbf{X}_B{}'\mathbf{X}_B\beta_1 + \beta_1{}'\mathbf{X}_B{}'\mathbf{X}_2\beta^{(2)} + \beta^{(2)\prime}\mathbf{X}_2{}'\mathbf{X}_B\beta_1}{a(r - 1)}$$

$$+ \frac{\beta^{(2)\prime}\mathbf{X}_2{}'\mathbf{X}_2\beta^{(2)} - \beta_1{}'\mathbf{X}_B{}'(\mathbf{I}_a\otimes r^{-1}\mathbf{J}_r\otimes b^{-1}\mathbf{J}_b)\mathbf{X}_B\beta_1}{a(r - 1)}$$

$$- \frac{\beta_1{}'\mathbf{X}_B{}'\mathbf{X}_2\beta^{(2)} + \beta^{(2)\prime}\mathbf{X}_2{}'\mathbf{X}_B\beta_1 + \beta^{(2)\prime}\mathbf{X}_2{}'\mathbf{X}_2\beta^{(2)}}{a(r - 1)}$$

$$- \frac{\beta_1{}'\mathbf{X}_B{}'[\mathbf{I}_a\otimes(\mathbf{I}_r-r^{-1}\mathbf{J}_r)\otimes b^{-1}\mathbf{J}_b]\mathbf{X}_B\beta_1}{a(r - 1)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

Within Classifications

Regression

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_6\mathbf{Y}}{tr(\mathbf{A}_6)}\right] = \frac{tr(\mathbf{A}_6\Sigma) + \mu'\mathbf{A}_6\mu}{tr(\mathbf{A}_6)}$$

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_6\mathbf{Y}}{tr(\mathbf{A}_6)}\right] = \sigma^2_{\varepsilon} + (\beta_1'\mathbf{X}_W' + \beta^{(2)}{}'\mathbf{X}_2')\,(\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W')\,(\mathbf{X}_W\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= \sigma^2_{\varepsilon} + \beta_1'\mathbf{X}_W'\mathbf{X}_W\beta_1$$

Residual

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_7\mathbf{Y}}{tr(\mathbf{A}_7)}\right] = \frac{tr(\mathbf{A}_7\Sigma) + \mu'\mathbf{A}_7\mu}{tr(\mathbf{A}_7)}$$

$$= \sigma^2_{\varepsilon} + \frac{(\beta_1'\mathbf{X}_W' + \beta^{(2)}{}'\mathbf{X}_2')\,(\mathbf{I}_a \otimes \mathbf{I}_r \otimes \mathbf{I}_b)\,(\mathbf{X}_W\beta_1 + \mathbf{X}_2\beta^{(2)})}{ar(b - 1) - 1}$$

$$- \frac{(\beta_1'\mathbf{X}_W' + \beta^{(2)}{}'\mathbf{X}_2')\,(\mathbf{I}_a \otimes \mathbf{I}_r \otimes b^{-1}\mathbf{J}_b)\,(\mathbf{X}_W\beta_1 + \mathbf{X}_2\beta^{(2)})}{ar(b - 1) - 1}$$

$$- \frac{(\beta_1'\mathbf{X}_W' + \beta^{(2)}{}'\mathbf{X}_2')\,(\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W')\,(\mathbf{X}_W\beta_1 + \mathbf{X}_2\beta^{(2)})}{ar(b - 1) - 1}$$

$$= \sigma^2_{\varepsilon} + \beta_1'\mathbf{X}_W'\mathbf{X}_W\beta_1 - \beta_1'\mathbf{X}_W'\mathbf{X}_W\beta_1 = \sigma^2_{\varepsilon}$$

Overall Regression

$$E\left[\frac{\mathbf{Y}'[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1']\mathbf{Y}}{tr[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1']}\right] = \frac{tr[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1\Sigma] + \mu'\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mu}{tr[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1']}$$

$$= b\sigma^2_R + \sigma^2_{\varepsilon} + (\beta_1'\mathbf{X}_1' + \beta^{(2)}{}'\mathbf{X}_2')\,(\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1')\,(\mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= b\sigma^2_R + \sigma^2_{\varepsilon} + \beta_1'\mathbf{X}_1'\mathbf{X}_1\beta_1 + \beta_1'\mathbf{X}_1'\mathbf{X}_2\beta^{(2)} + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_1\beta_1 + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

Expected Mean Squares for Table II

Between Classifications

Mean

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_1\mathbf{Y}}{tr(\mathbf{A}_1)}\right] = \frac{tr(\mathbf{A}_1\Sigma) + \mu'\mathbf{A}_1\mu}{tr(\mathbf{A}_1)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + (\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(a^{-1}\mathbf{J}_a \otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

Regression

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_2\mathbf{Y}}{tr(\mathbf{A}_2)}\right] = \frac{tr(\mathbf{A}_2\Sigma) + \mu'\mathbf{A}_2\mu}{tr(\mathbf{A}_2)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + (\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B')(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \beta_1'\mathbf{X}_B'\mathbf{X}_B\beta_1 + \beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)} + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B\beta_1 + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_2\beta^{(2)}$$

Residual

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_3\mathbf{Y}}{tr(\mathbf{A}_3)}\right] = \frac{tr(\mathbf{A}_3\Sigma) + \mu'\mathbf{A}_3\mu}{tr(\mathbf{A}_3)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \frac{(\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{I}_a \otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})}{a - 2}$$

$$- \frac{(\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(a^{-1}\mathbf{J}_a \otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})}{a - 2}$$

$$- \frac{(\beta_1'\mathbf{X}_B' + \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B')(\mathbf{X}_B\beta_1 + \mathbf{X}_2\beta^{(2)})}{a - 2}$$

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_3\mathbf{Y}}{tr(\mathbf{A}_3)}\right] = b\sigma_R^2 + \sigma_\varepsilon^2 + \frac{\beta_1'\mathbf{X}_B'\mathbf{X}_B\beta_1 + \beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{a - 2}$$

$$+ \frac{\beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_B\beta_1 + \beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_2\beta^{(2)} - \beta_1'\mathbf{X}_B'\mathbf{X}_B\beta_1}{a - 2}$$

$$- \frac{\beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)} + \beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_B\beta_1}{a - 2}$$

$$- \frac{\beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{a - 2}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

$$+ \frac{\beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_2\beta^{(2)} - \beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_B(\mathbf{X}_B'\mathbf{X}_B)^{-1}\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{a - 2}$$

Within Classifications

Regression

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_4\mathbf{Y}}{tr(\mathbf{A}_4)}\right] = \frac{tr(\mathbf{A}_4\Sigma) + \mu'\mathbf{A}_4\mu}{tr(\mathbf{A}_4)}$$

$$= \sigma_\varepsilon^2 + (\beta_1'\mathbf{X}_W' + \beta^{(2)\prime}\mathbf{X}_2')(\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W')(\mathbf{X}_W\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= \sigma_\varepsilon^2 + \beta_1'\mathbf{X}_W'\mathbf{X}_W\beta_1$$

Residual

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_5\mathbf{Y}}{tr(\mathbf{A}_5)}\right] = \frac{tr(\mathbf{A}_5\Sigma) + \mu'\mathbf{A}_5\mu}{tr(\mathbf{A}_5)}$$

$$= \sigma_\varepsilon^2 + \frac{(\beta_1'\mathbf{X}_W' + \beta^{(2)\prime}\mathbf{X}_2')(\mathbf{I}_a \otimes \mathbf{I}_b)(\mathbf{X}_W\beta_1 + \mathbf{X}_2\beta^{(2)})}{a(b - 1) - 1}$$

$$- \frac{(\beta_1'\mathbf{X}_W' + \beta^{(2)\prime}\mathbf{X}_2')(\mathbf{I}_a \otimes b^{-1}\mathbf{J}_b)(\mathbf{X}_W\beta_1 + \mathbf{X}_2\beta^{(2)})}{a(b - 1) - 1}$$

$$- \frac{(\beta_1'\mathbf{X}_W' + \beta^{(2)\prime}\mathbf{X}_2')(\mathbf{X}_W(\mathbf{X}_W'\mathbf{X}_W)^{-1}\mathbf{X}_W')(\mathbf{X}_W\beta_1 + \mathbf{X}_2\beta^{(2)})}{a(b - 1) - 1}$$

$$= \sigma_\varepsilon^2 + \beta_1'\mathbf{X}_W'\mathbf{X}_W\beta_1 - \beta_1'\mathbf{X}_W'\mathbf{X}_W\beta_1 = \sigma_\varepsilon^2$$

Overall Regression

$$E\left(\frac{\mathbf{Y}'[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1']\mathbf{Y}}{\mathrm{tr}[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1']}\right) = \frac{\mathrm{tr}[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1\Sigma] + \mu'\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mu}{\mathrm{tr}[\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1']}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + (\beta_1'\mathbf{X}_1' + \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1')(\mathbf{X}_1\beta_1 + \mathbf{X}_2\beta^{(2)})$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \beta_1'\mathbf{X}_1'\mathbf{X}_1\beta_1 + \beta_1'\mathbf{X}_1'\mathbf{X}_2\beta^{(2)} + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_1\beta_1 + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_1(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathbf{X}_2\beta^{(2)}$$

Expected Mean Squares for Table IV

Between Classifications

Mean

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_1\mathbf{Y}}{tr(\mathbf{A}_1)}\right] = \frac{tr(\mathbf{A}_1\Sigma) + \mu'\mathbf{A}_1\mu}{tr(\mathbf{A}_1)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

$$+ (\beta_1'\mathbf{X}_B' \quad \beta^{(2)\prime}\mathbf{X}_2')(a^{-1}\mathbf{J}_a \otimes r^{-1}\mathbf{J}_r \otimes b^{-1}\mathbf{J}_b)\begin{bmatrix}\mathbf{X}_B\beta_1\\ \mathbf{X}_2\beta^{(2)}\end{bmatrix}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

Regression

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_2\mathbf{Y}}{tr(\mathbf{A}_2)}\right] = \frac{tr(\mathbf{A}_2\Sigma) + \mu'\mathbf{A}_2\mu}{tr(\mathbf{A}_2)}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

$$+ (\beta_1'\mathbf{X}_B' \quad \beta^{(2)\prime}\mathbf{X}_2')(\mathbf{X}_B \quad \mathbf{X}_2)\left[\begin{bmatrix}\mathbf{X}_B'\\ \mathbf{X}_2'\end{bmatrix}(\mathbf{X}_B \quad \mathbf{X}_2)\right]^{-1}\begin{bmatrix}\mathbf{X}_B'\\ \mathbf{X}_2'\end{bmatrix}\begin{bmatrix}\mathbf{X}_B\beta_1\\ \mathbf{X}_2\beta^{(2)}\end{bmatrix}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + (\beta_1'\mathbf{X}_B' \quad \beta^{(2)\prime}\mathbf{X}_2')(\mathbf{X}_B \quad \mathbf{X}_2)\begin{bmatrix}\mathbf{X}_B'\\ \mathbf{X}_2'\end{bmatrix}\begin{bmatrix}\mathbf{X}_B\beta_1\\ \mathbf{X}_2\beta^{(2)}\end{bmatrix}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \beta_1'\mathbf{X}_B'\mathbf{X}_B\beta_1 + \beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)} + \beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_B\beta_1 + \beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_2\beta^{(2)}$$

Residual

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_3\mathbf{Y}}{tr(\mathbf{A}_3)}\right] = \frac{tr(\mathbf{A}_3\Sigma) + \mu'\mathbf{A}_3\mu}{tr(\mathbf{A}_3)}$$

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_3\mathbf{Y}}{tr(\mathbf{A}_3)}\right] = b\sigma_R^2 + \sigma_\varepsilon^2$$

$$+ \frac{(\beta_1'\mathbf{X}_B' \quad \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{I}_a \otimes \mathbf{I}_r \otimes b^{-1}\mathbf{J}_b)\begin{pmatrix}\mathbf{X}_B\beta_1 \\ \mathbf{X}_2\beta^{(2)}\end{pmatrix}}{ar - 2}$$

$$- \frac{(\beta_1'\mathbf{X}_B' \quad \beta^{(2)}{}'\mathbf{X}_2')(a^{-1}\mathbf{J}_a \otimes r^{-1}\mathbf{J}_r \otimes b^{-1}\mathbf{J}_b)\begin{pmatrix}\mathbf{X}_B\beta_1 \\ \mathbf{X}_2\beta^{(2)}\end{pmatrix}}{ar - 2}$$

$$- \frac{(\beta_1'\mathbf{X}_B' \quad \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{X}_{B2}(\mathbf{X}_{B2}'\mathbf{X}_{B2})^{-1}\mathbf{X}_{B2}')\begin{pmatrix}\mathbf{X}_B\beta_1 \\ \mathbf{X}_2\beta^{(2)}\end{pmatrix}}{ar - 2}$$

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_3\mathbf{Y}}{tr(\mathbf{A}_3)}\right] = b\sigma_R^2 + \sigma_\varepsilon^2 + \frac{\beta_1'\mathbf{X}_B'\mathbf{X}_B\beta_1 - \beta_1'\mathbf{X}_B'\mathbf{X}_B\beta_1 + \beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)}}{ar - 2}$$

$$- \frac{\beta_1'\mathbf{X}_B'\mathbf{X}_2\beta^{(2)} - \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B\beta_1 + \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_B\beta_1}{ar - 2}$$

$$+ \frac{\beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_2\beta^{(2)} - \beta^{(2)}{}'\mathbf{X}_2'\mathbf{X}_2\beta^{(2)}}{ar - 2}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2$$

<u>Within Classifications</u>

<u>Regression</u>

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_4\mathbf{Y}}{tr(\mathbf{A}_4)}\right] = \frac{tr(\mathbf{A}_4\boldsymbol{\Sigma}) + \mu'\mathbf{A}_4\mu}{tr(\mathbf{A}_4)}$$

$$= \sigma_\varepsilon^2$$

$$+ (\beta_1'\mathbf{X}_W' \quad \beta^{(2)}{}'\mathbf{X}_2')(\mathbf{X}_W \quad \mathbf{0})\left[\begin{pmatrix}\mathbf{X}_W' \\ \mathbf{0}\end{pmatrix}(\mathbf{X}_W \quad \mathbf{0})\right]^{-1}\begin{pmatrix}\mathbf{X}_W' \\ \mathbf{0}\end{pmatrix}\begin{pmatrix}\mathbf{X}_W\beta_1 \\ \mathbf{X}_2\beta^{(2)}\end{pmatrix}$$

$$= \sigma_\varepsilon^2 + \beta_1'\mathbf{X}_W'\mathbf{X}_W'\beta_1$$

Residual

$$E\left[\frac{\mathbf{Y}'\mathbf{A}_5\mathbf{Y}}{\mathrm{tr}(\mathbf{A}_5)}\right] = \frac{\mathrm{tr}(\mathbf{A}_5\Sigma) + \mu'\mathbf{A}_5\mu}{\mathrm{tr}(\mathbf{A}_5)}$$

$$= \sigma_\varepsilon^2 + \frac{(\beta_1'\mathbf{X}_W' \quad \beta^{(2)\prime}\mathbf{X}_2')(\mathbf{I}_a\otimes\mathbf{I}_r\otimes\mathbf{I}_b)\begin{bmatrix}\mathbf{X}_W\beta_1\\ \mathbf{X}_2\beta^{(2)}\end{bmatrix}}{ar(b-1)-1}$$

$$- \frac{(\beta_1'\mathbf{X}_W' \quad \beta^{(2)\prime}\mathbf{X}_2')(a^{-1}\mathbf{J}_a\otimes r^{-1}\mathbf{J}_r\otimes b^{-1}\mathbf{J}_b)\begin{bmatrix}\mathbf{X}_W\beta_1\\ \mathbf{X}_2\beta^{(2)}\end{bmatrix}}{ar(b-1)-1}$$

$$- \frac{(\beta_1'\mathbf{X}_W' \quad \beta^{(2)\prime}\mathbf{X}_2')(\mathbf{X}_{W2}(\mathbf{X}_{W2}'\mathbf{X}_{W2})^{-1}\mathbf{X}_{W2}')\begin{bmatrix}\mathbf{X}_W\beta_1\\ \mathbf{X}_2\beta^{(2)}\end{bmatrix}}{ar(b-1)-1}$$

$$= \sigma_\varepsilon^2 + \beta_1'\mathbf{X}_W'\mathbf{X}_W\beta_1 - \beta_1'\mathbf{X}_W'\mathbf{X}_W\beta_1 = \sigma_\varepsilon^2$$

Overall Regression

For $\mathbf{X} = (\mathbf{X}_1 \quad \mathbf{X}_2)$

$$E\left[\frac{\mathbf{Y}'[\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}']\mathbf{Y}}{\mathrm{tr}[\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}']}\right] = \frac{\mathrm{tr}[\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}\Sigma] + \mu'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mu}{\mathrm{tr}[\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}']}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + (\beta_1'\mathbf{X}_B' \quad \beta^{(2)\prime}\mathbf{X}_2')(\mathbf{X}_1 \quad \mathbf{X}_2)\begin{bmatrix}\mathbf{X}_1'\\ \mathbf{X}_2'\end{bmatrix}\begin{bmatrix}\mathbf{X}_1\beta_1\\ \mathbf{X}_2\beta^{(2)}\end{bmatrix}$$

$$= b\sigma_R^2 + \sigma_\varepsilon^2 + \beta_1'\mathbf{X}_1'\mathbf{X}_1\beta_1 + \beta_1'\mathbf{X}_1'\mathbf{X}_2\beta^{(2)} + \beta^{(2)\prime}\mathbf{X}_1'\mathbf{X}_B\beta_1 + \beta^{(2)\prime}\mathbf{X}_2'\mathbf{X}_2\beta^{(2)}$$

APPENDIX B
RELATIVE EFFICIENCIES FOR GIVEN PARAMETER VALUES OF λ, γ, ρ, AND b

SAS Program

```
OPTIONS PS = 60;

DATA RELEFF;
   DO B = 2, 3, 6, 12;
      DO LAMBDA = 0, .5, 1, 2, 5;
         DO RHO = 0, .5, 1, 2, 5;
            DO GAMMA = .001, .2, .5, .7, .99;
               RELEFF =
                ((GAMMA*(1+B*RHO)*(1+B*RHO*GAMMA))+LAMBDA)/
                          (1+B*RHO*GAMMA)**2;
               OUTPUT;
            END;
         END;
      END;
   END;

PROC PRINT DATA = RELEFF NOOBS;
RUN;
```

Output

B	LAMBDA	RHO	GAMMA	RELEFF
2	0.0	0.0	0.001	0.00100
2	0.0	0.0	0.200	0.20000
2	0.0	0.0	0.500	0.50000
2	0.0	0.0	0.700	0.70000
2	0.0	0.0	0.990	0.99000
2	0.0	0.5	0.001	0.00200
2	0.0	0.5	0.200	0.33333
2	0.0	0.5	0.500	0.66667
2	0.0	0.5	0.700	0.82353
2	0.0	0.5	0.990	0.99497
2	0.0	1.0	0.001	0.00299
2	0.0	1.0	0.200	0.42857
2	0.0	1.0	0.500	0.75000
2	0.0	1.0	0.700	0.87500
2	0.0	1.0	0.990	0.99664
2	0.0	2.0	0.001	0.00498
2	0.0	2.0	0.200	0.55556

B	LAMBDA	RHO	GAMMA	RELEFF
2	0.0	2.0	0.500	0.83333
2	0.0	2.0	0.700	0.92105
2	0.0	2.0	0.990	0.99798
2	0.0	5.0	0.001	0.01089
2	0.0	5.0	0.200	0.73333
2	0.0	5.0	0.500	0.91667
2	0.0	5.0	0.700	0.96250
2	0.0	5.0	0.990	0.99908
2	0.5	0.0	0.001	0.50100
2	0.5	0.0	0.200	0.70000
2	0.5	0.0	0.500	1.00000
2	0.5	0.0	0.700	1.20000
2	0.5	0.0	0.990	1.49000
2	0.5	0.5	0.001	0.50100
2	0.5	0.5	0.200	0.68056
2	0.5	0.5	0.500	0.88889
2	0.5	0.5	0.700	0.99654
2	0.5	0.5	0.990	1.12123
2	0.5	1.0	0.001	0.50100
2	0.5	1.0	0.200	0.68367
2	0.5	1.0	0.500	0.87500
2	0.5	1.0	0.700	0.96181
2	0.5	1.0	0.990	1.05295
2	0.5	2.0	0.001	0.50100
2	0.5	2.0	0.200	0.70988
2	0.5	2.0	0.500	0.88889
2	0.5	2.0	0.700	0.95568
2	0.5	2.0	0.990	1.01831
2	0.5	5.0	0.001	0.50104
2	0.5	5.0	0.200	0.78889
2	0.5	5.0	0.500	0.93056
2	0.5	5.0	0.700	0.97031
2	0.5	5.0	0.990	1.00329
2	1.0	0.0	0.001	1.00100
2	1.0	0.0	0.200	1.20000
2	1.0	0.0	0.500	1.50000
2	1.0	0.0	0.700	1.70000
2	1.0	0.0	0.990	1.99000
2	1.0	0.5	0.001	1.00000
2	1.0	0.5	0.200	1.02778
2	1.0	0.5	0.500	1.11111
2	1.0	0.5	0.700	1.16955
2	1.0	0.5	0.990	1.24749
2	1.0	1.0	0.001	0.99901
2	1.0	1.0	0.200	0.93878
2	1.0	1.0	0.500	1.00000
2	1.0	1.0	0.700	1.04861
2	1.0	1.0	0.990	1.10925
2	1.0	2.0	0.001	0.99703
2	1.0	2.0	0.200	0.86420
2	1.0	2.0	0.500	0.94444
2	1.0	2.0	0.700	0.99030

B	LAMBDA	RHO	GAMMA	RELEFF
2	1.0	2.0	0.990	1.03863
2	1.0	5.0	0.001	0.99119
2	1.0	5.0	0.200	0.84444
2	1.0	5.0	0.500	0.94444
2	1.0	5.0	0.700	0.97813
2	1.0	5.0	0.990	1.00750
2	2.0	0.0	0.001	2.00100
2	2.0	0.0	0.200	2.20000
2	2.0	0.0	0.500	2.50000
2	2.0	0.0	0.700	2.70000
2	2.0	0.0	0.990	2.99000
2	2.0	0.5	0.001	1.99800
2	2.0	0.5	0.200	1.72222
2	2.0	0.5	0.500	1.55556
2	2.0	0.5	0.700	1.51557
2	2.0	0.5	0.990	1.50001
2	2.0	1.0	0.001	1.99502
2	2.0	1.0	0.200	1.44898
2	2.0	1.0	0.500	1.25000
2	2.0	1.0	0.700	1.22222
2	2.0	1.0	0.990	1.22186
2	2.0	2.0	0.001	1.98908
2	2.0	2.0	0.200	1.17284
2	2.0	2.0	0.500	1.05556
2	2.0	2.0	0.700	1.05956
2	2.0	2.0	0.990	1.07928
2	2.0	5.0	0.001	1.97148
2	2.0	5.0	0.200	0.95556
2	2.0	5.0	0.500	0.97222
2	2.0	5.0	0.700	0.99375
2	2.0	5.0	0.990	1.01592
2	5.0	0.0	0.001	5.00100
2	5.0	0.0	0.200	5.20000
2	5.0	0.0	0.500	5.50000
2	5.0	0.0	0.700	5.70000
2	5.0	0.0	0.990	5.99000
2	5.0	0.5	0.001	4.99201
2	5.0	0.5	0.200	3.80556
2	5.0	0.5	0.500	2.88889
2	5.0	0.5	0.700	2.55363
2	5.0	0.5	0.990	2.25757
2	5.0	1.0	0.001	4.98305
2	5.0	1.0	0.200	2.97959
2	5.0	1.0	0.500	2.00000
2	5.0	1.0	0.700	1.74306
2	5.0	1.0	0.990	1.55968
2	5.0	2.0	0.001	4.96522
2	5.0	2.0	0.200	2.09877
2	5.0	2.0	0.500	1.38889
2	5.0	2.0	0.700	1.26731
2	5.0	2.0	0.990	1.20122
2	5.0	5.0	0.001	4.91237

B	LAMBDA	RHO	GAMMA	RELEFF
2	5.0	5.0	0.200	1.28889
2	5.0	5.0	0.500	1.05556
2	5.0	5.0	0.700	1.04063
2	5.0	5.0	0.990	1.04117
3	0.0	0.0	0.001	0.00100
3	0.0	0.0	0.200	0.20000
3	0.0	0.0	0.500	0.50000
3	0.0	0.0	0.700	0.70000
3	0.0	0.0	0.990	0.99000
3	0.0	0.5	0.001	0.00250
3	0.0	0.5	0.200	0.38462
3	0.0	0.5	0.500	0.71429
3	0.0	0.5	0.700	0.85366
3	0.0	0.5	0.990	0.99598
3	0.0	1.0	0.001	0.00399
3	0.0	1.0	0.200	0.50000
3	0.0	1.0	0.500	0.80000
3	0.0	1.0	0.700	0.90323
3	0.0	1.0	0.990	0.99748
3	0.0	2.0	0.001	0.00696
3	0.0	2.0	0.200	0.63636
3	0.0	2.0	0.500	0.87500
3	0.0	2.0	0.700	0.94231
3	0.0	2.0	0.990	0.99856
3	0.0	5.0	0.001	0.01576
3	0.0	5.0	0.200	0.80000
3	0.0	5.0	0.500	0.94118
3	0.0	5.0	0.700	0.97391
3	0.0	5.0	0.990	0.99937
3	0.5	0.0	0.001	0.50100
3	0.5	0.0	0.200	0.70000
3	0.5	0.0	0.500	1.00000
3	0.5	0.0	0.700	1.20000
3	0.5	0.0	0.990	1.49000
3	0.5	0.5	0.001	0.50100
3	0.5	0.5	0.200	0.68047
3	0.5	0.5	0.500	0.87755
3	0.5	0.5	0.700	0.97264
3	0.5	0.5	0.990	1.07694
3	0.5	1.0	0.001	0.50100
3	0.5	1.0	0.200	0.69531
3	0.5	1.0	0.500	0.88000
3	0.5	1.0	0.700	0.95525
3	0.5	1.0	0.990	1.02921
3	0.5	2.0	0.001	0.50101
3	0.5	2.0	0.200	0.73967
3	0.5	2.0	0.500	0.90625
3	0.5	2.0	0.700	0.96080
3	0.5	2.0	0.990	1.00894
3	0.5	5.0	0.001	0.50109
3	0.5	5.0	0.200	0.83125
3	0.5	5.0	0.500	0.94810

B	LAMBDA	RHO	GAMMA	RELEFF
3	0.5	5.0	0.700	0.97769
3	0.5	5.0	0.990	1.00136
3	1.0	0.0	0.001	1.00100
3	1.0	0.0	0.200	1.20000
3	1.0	0.0	0.500	1.50000
3	1.0	0.0	0.700	1.70000
3	1.0	0.0	0.990	1.99000
3	1.0	0.5	0.001	0.99950
3	1.0	0.5	0.200	0.97633
3	1.0	0.5	0.500	1.04082
3	1.0	0.5	0.700	1.09161
3	1.0	0.5	0.990	1.15791
3	1.0	1.0	0.001	0.99801
3	1.0	1.0	0.200	0.89063
3	1.0	1.0	0.500	0.96000
3	1.0	1.0	0.700	1.00728
3	1.0	1.0	0.990	1.06093
3	1.0	2.0	0.001	0.99507
3	1.0	2.0	0.200	0.84298
3	1.0	2.0	0.500	0.93750
3	1.0	2.0	0.700	0.97929
3	1.0	2.0	0.990	1.01932
3	1.0	5.0	0.001	0.98643
3	1.0	5.0	0.200	0.86250
3	1.0	5.0	0.500	0.95502
3	1.0	5.0	0.700	0.98147
3	1.0	5.0	0.990	1.00335
3	2.0	0.0	0.001	2.00100
3	2.0	0.0	0.200	2.20000
3	2.0	0.0	0.500	2.50000
3	2.0	0.0	0.700	2.70000
3	2.0	0.0	0.990	2.99000
3	2.0	0.5	0.001	1.99651
3	2.0	0.5	0.200	1.56805
3	2.0	0.5	0.500	1.36735
3	2.0	0.5	0.700	1.32957
3	2.0	0.5	0.990	1.31985
3	2.0	1.0	0.001	1.99204
3	2.0	1.0	0.200	1.28125
3	2.0	1.0	0.500	1.12000
3	2.0	1.0	0.700	1.11134
3	2.0	1.0	0.990	1.12438
3	2.0	2.0	0.001	1.98317
3	2.0	2.0	0.200	1.04959
3	2.0	2.0	0.500	1.00000
3	2.0	2.0	0.700	1.01627
3	2.0	2.0	0.990	1.04008
3	2.0	5.0	0.001	1.95709
3	2.0	5.0	0.200	0.92500
3	2.0	5.0	0.500	0.96886
3	2.0	5.0	0.700	0.98904
3	2.0	5.0	0.990	1.00733

B	LAMBDA	RHO	GAMMA	RELEFF
3	5.0	0.0	0.001	5.00100
3	5.0	0.0	0.200	5.20000
3	5.0	0.0	0.500	5.50000
3	5.0	0.0	0.700	5.70000
3	5.0	0.0	0.990	5.99000
3	5.0	0.5	0.001	4.98753
3	5.0	0.5	0.200	3.34320
3	5.0	0.5	0.500	2.34694
3	5.0	0.5	0.700	2.04343
3	5.0	0.5	0.990	1.80566
3	5.0	1.0	0.001	4.97412
3	5.0	1.0	0.200	2.45313
3	5.0	1.0	0.500	1.60000
3	5.0	1.0	0.700	1.42352
3	5.0	1.0	0.990	1.31472
3	5.0	2.0	0.001	4.94749
3	5.0	2.0	0.200	1.66942
3	5.0	2.0	0.500	1.18750
3	5.0	2.0	0.700	1.12722
3	5.0	2.0	0.990	1.10237
3	5.0	5.0	0.001	4.86907
3	5.0	5.0	0.200	1.11250
3	5.0	5.0	0.500	1.01038
3	5.0	5.0	0.700	1.01172
3	5.0	5.0	0.990	1.01927
6	0.0	0.0	0.001	0.00100
6	0.0	0.0	0.200	0.20000
6	0.0	0.0	0.500	0.50000
6	0.0	0.0	0.700	0.70000
6	0.0	0.0	0.990	0.99000
6	0.0	0.5	0.001	0.00399
6	0.0	0.5	0.200	0.50000
6	0.0	0.5	0.500	0.80000
6	0.0	0.5	0.700	0.90323
6	0.0	0.5	0.990	0.99748
6	0.0	1.0	0.001	0.00696
6	0.0	1.0	0.200	0.63636
6	0.0	1.0	0.500	0.87500
6	0.0	1.0	0.700	0.94231
6	0.0	1.0	0.990	0.99856
6	0.0	2.0	0.001	0.01285
6	0.0	2.0	0.200	0.76471
6	0.0	2.0	0.500	0.92857
6	0.0	2.0	0.700	0.96809
6	0.0	2.0	0.990	0.99922
6	0.0	5.0	0.001	0.03010
6	0.0	5.0	0.200	0.88571
6	0.0	5.0	0.500	0.96875
6	0.0	5.0	0.700	0.98636
6	0.0	5.0	0.990	0.99967
6	0.5	0.0	0.001	0.50100
6	0.5	0.0	0.200	0.70000

B	LAMBDA	RHO	GAMMA	RELEFF
6	0.5	0.0	0.500	1.00000
6	0.5	0.0	0.700	1.20000
6	0.5	0.0	0.990	1.49000
6	0.5	0.5	0.001	0.50100
6	0.5	0.5	0.200	0.69531
6	0.5	0.5	0.500	0.88000
6	0.5	0.5	0.700	0.95525
6	0.5	0.5	0.990	1.02921
6	0.5	1.0	0.001	0.50101
6	0.5	1.0	0.200	0.73967
6	0.5	1.0	0.500	0.90625
6	0.5	1.0	0.700	0.96080
6	0.5	1.0	0.990	1.00894
6	0.5	2.0	0.001	0.50106
6	0.5	2.0	0.200	0.80796
6	0.5	2.0	0.500	0.93878
6	0.5	2.0	0.700	0.97374
6	0.5	2.0	0.990	1.00224
6	0.5	5.0	0.001	0.50140
6	0.5	5.0	0.200	0.89592
6	0.5	5.0	0.500	0.97070
6	0.5	5.0	0.700	0.98740
6	0.5	5.0	0.990	1.00020
6	1.0	0.0	0.001	1.00100
6	1.0	0.0	0.200	1.20000
6	1.0	0.0	0.500	1.50000
6	1.0	0.0	0.700	1.70000
6	1.0	0.0	0.990	1.99000
6	1.0	0.5	0.001	0.99801
6	1.0	0.5	0.200	0.89063
6	1.0	0.5	0.500	0.96000
6	1.0	0.5	0.700	1.00728
6	1.0	0.5	0.990	1.06093
6	1.0	1.0	0.001	0.99507
6	1.0	1.0	0.200	0.84298
6	1.0	1.0	0.500	0.93750
6	1.0	1.0	0.700	0.97929
6	1.0	1.0	0.990	1.01932
6	1.0	2.0	0.001	0.98927
6	1.0	2.0	0.200	0.85121
6	1.0	2.0	0.500	0.94898
6	1.0	2.0	0.700	0.97940
6	1.0	2.0	0.990	1.00525
6	1.0	5.0	0.001	0.97269
6	1.0	5.0	0.200	0.90612
6	1.0	5.0	0.500	0.97266
6	1.0	5.0	0.700	0.98843
6	1.0	5.0	0.990	1.00074
6	2.0	0.0	0.001	2.00100
6	2.0	0.0	0.200	2.20000
6	2.0	0.0	0.500	2.50000
6	2.0	0.0	0.700	2.70000

B	LAMBDA	RHO	GAMMA	RELEFF
6	2.0	0.0	0.990	2.99000
6	2.0	0.5	0.001	1.99204
6	2.0	0.5	0.200	1.28125
6	2.0	0.5	0.500	1.12000
6	2.0	0.5	0.700	1.11134
6	2.0	0.5	0.990	1.12438
6	2.0	1.0	0.001	1.98317
6	2.0	1.0	0.200	1.04959
6	2.0	1.0	0.500	1.00000
6	2.0	1.0	0.700	1.01627
6	2.0	1.0	0.990	1.04008
6	2.0	2.0	0.001	1.96570
6	2.0	2.0	0.200	0.93772
6	2.0	2.0	0.500	0.96939
6	2.0	2.0	0.700	0.99072
6	2.0	2.0	0.990	1.01128
6	2.0	5.0	0.001	1.91529
6	2.0	5.0	0.200	0.92653
6	2.0	5.0	0.500	0.97656
6	2.0	5.0	0.700	0.99050
6	2.0	5.0	0.990	1.00180
6	5.0	0.0	0.001	5.00100
6	5.0	0.0	0.200	5.20000
6	5.0	0.0	0.500	5.50000
6	5.0	0.0	0.700	5.70000
6	5.0	0.0	0.990	5.99000
6	5.0	0.5	0.001	4.97412
6	5.0	0.5	0.200	2.45313
6	5.0	0.5	0.500	1.60000
6	5.0	0.5	0.700	1.42352
6	5.0	0.5	0.990	1.31472
6	5.0	1.0	0.001	4.94749
6	5.0	1.0	0.200	1.66942
6	5.0	1.0	0.500	1.18750
6	5.0	1.0	0.700	1.12722
6	5.0	1.0	0.990	1.10237
6	5.0	2.0	0.001	4.89497
6	5.0	2.0	0.200	1.19723
6	5.0	2.0	0.500	1.03061
6	5.0	2.0	0.700	1.02467
6	5.0	2.0	0.990	1.02936
6	5.0	5.0	0.001	4.74308
6	5.0	5.0	0.200	0.98776
6	5.0	5.0	0.500	0.98828
6	5.0	5.0	0.700	0.99669
6	5.0	5.0	0.990	1.00498
12	0.0	0.0	0.001	0.00100
12	0.0	0.0	0.200	0.20000
12	0.0	0.0	0.500	0.50000
12	0.0	0.0	0.700	0.70000
12	0.0	0.0	0.990	0.99000
12	0.0	0.5	0.001	0.00696

B	LAMBDA	RHO	GAMMA	RELEFF
12	0.0	0.5	0.200	0.63636
12	0.0	0.5	0.500	0.87500
12	0.0	0.5	0.700	0.94231
12	0.0	0.5	0.990	0.99856
12	0.0	1.0	0.001	0.01285
12	0.0	1.0	0.200	0.76471
12	0.0	1.0	0.500	0.92857
12	0.0	1.0	0.700	0.96809
12	0.0	1.0	0.990	0.99922
12	0.0	2.0	0.001	0.02441
12	0.0	2.0	0.200	0.86207
12	0.0	2.0	0.500	0.96154
12	0.0	2.0	0.700	0.98315
12	0.0	2.0	0.990	0.99960
12	0.0	5.0	0.001	0.05755
12	0.0	5.0	0.200	0.93846
12	0.0	5.0	0.500	0.98387
12	0.0	5.0	0.700	0.99302
12	0.0	5.0	0.990	0.99983
12	0.5	0.0	0.001	0.50100
12	0.5	0.0	0.200	0.70000
12	0.5	0.0	0.500	1.00000
12	0.5	0.0	0.700	1.20000
12	0.5	0.0	0.990	1.49000
12	0.5	0.5	0.001	0.50101
12	0.5	0.5	0.200	0.73967
12	0.5	0.5	0.500	0.90625
12	0.5	0.5	0.700	0.96080
12	0.5	0.5	0.990	1.00894
12	0.5	1.0	0.001	0.50106
12	0.5	1.0	0.200	0.80796
12	0.5	1.0	0.500	0.93878
12	0.5	1.0	0.700	0.97374
12	0.5	1.0	0.990	1.00224
12	0.5	2.0	0.001	0.50125
12	0.5	2.0	0.200	0.87693
12	0.5	2.0	0.500	0.96450
12	0.5	2.0	0.700	0.98472
12	0.5	2.0	0.990	1.00041
12	0.5	5.0	0.001	0.50255
12	0.5	5.0	0.200	0.94142
12	0.5	5.0	0.500	0.98439
12	0.5	5.0	0.700	0.99329
12	0.5	5.0	0.990	0.99997
12	1.0	0.0	0.001	1.00100
12	1.0	0.0	0.200	1.20000
12	1.0	0.0	0.500	1.50000
12	1.0	0.0	0.700	1.70000
12	1.0	0.0	0.990	1.99000
12	1.0	0.5	0.001	0.99507
12	1.0	0.5	0.200	0.84298
12	1.0	0.5	0.500	0.93750

B	LAMBDA	RHO	GAMMA	RELEFF
12	1.0	0.5	0.700	0.97929
12	1.0	0.5	0.990	1.01932
12	1.0	1.0	0.001	0.98927
12	1.0	1.0	0.200	0.85121
12	1.0	1.0	0.500	0.94898
12	1.0	1.0	0.700	0.97940
12	1.0	1.0	0.990	1.00525
12	1.0	2.0	0.001	0.97809
12	1.0	2.0	0.200	0.89180
12	1.0	2.0	0.500	0.96746
12	1.0	2.0	0.700	0.98630
12	1.0	2.0	0.990	1.00123
12	1.0	5.0	0.001	0.94754
12	1.0	5.0	0.200	0.94438
12	1.0	5.0	0.500	0.98491
12	1.0	5.0	0.700	0.99356
12	1.0	5.0	0.990	1.00011
12	2.0	0.0	0.001	2.00100
12	2.0	0.0	0.200	2.20000
12	2.0	0.0	0.500	2.50000
12	2.0	0.0	0.700	2.70000
12	2.0	0.0	0.990	2.99000
12	2.0	0.5	0.001	1.98317
12	2.0	0.5	0.200	1.04959
12	2.0	0.5	0.500	1.00000
12	2.0	0.5	0.700	1.01627
12	2.0	0.5	0.990	1.04008
12	2.0	1.0	0.001	1.96570
12	2.0	1.0	0.200	0.93772
12	2.0	1.0	0.500	0.96939
12	2.0	1.0	0.700	0.99072
12	2.0	1.0	0.990	1.01128
12	2.0	2.0	0.001	1.93176
12	2.0	2.0	0.200	0.92152
12	2.0	2.0	0.500	0.97337
12	2.0	2.0	0.700	0.98946
12	2.0	2.0	0.990	1.00286
12	2.0	5.0	0.001	1.83754
12	2.0	5.0	0.200	0.95030
12	2.0	5.0	0.500	0.98595
12	2.0	5.0	0.700	0.99410
12	2.0	5.0	0.990	1.00038
12	5.0	0.0	0.001	5.00100
12	5.0	0.0	0.200	5.20000
12	5.0	0.0	0.500	5.50000
12	5.0	0.0	0.700	5.70000
12	5.0	0.0	0.990	5.99000
12	5.0	0.5	0.001	4.94749
12	5.0	0.5	0.200	1.66942
12	5.0	0.5	0.500	1.18750
12	5.0	0.5	0.700	1.12722
12	5.0	0.5	0.990	1.10237

B	LAMBDA	RHO	GAMMA	RELEFF
12	5.0	1.0	0.001	4.89497
12	5.0	1.0	0.200	1.19723
12	5.0	1.0	0.500	1.03061
12	5.0	1.0	0.700	1.02467
12	5.0	1.0	0.990	1.02936
12	5.0	2.0	0.001	4.79279
12	5.0	2.0	0.200	1.01070
12	5.0	2.0	0.500	0.99112
12	5.0	2.0	0.700	0.99893
12	5.0	2.0	0.990	1.00775
12	5.0	5.0	0.001	4.50753
12	5.0	5.0	0.200	0.96805
12	5.0	5.0	0.500	0.98907
12	5.0	5.0	0.700	0.99573
12	5.0	5.0	0.990	1.00120

APPENDIX C
RELATIVE EFFICIENCIES FOR GIVEN PARAMETER VALUES
OF λ, γ, AND b FOR $\rho = 0$

SAS Program

```
OPTIONS PS = 60;

DATA RELEFF;
   DO B = 2, 3, 6, 12;
      DO LAMBDA = 0, .5, 1, 2, 5;
         DO RHO = 0;
            DO GAMMA = .001, .2, .5, .7, .99;
               RELEFF =
               ((GAMMA*(1+B*RHO)*(1+B*RHO*GAMMA))+LAMBDA)/
                         (1+B*RHO*GAMMA)**2;
               OUTPUT;
            END;
         END;
      END;
   END;

PROC PRINT DATA = RELEFF NOOBS;
RUN;
```

Output

B	LAMBDA	RHO	GAMMA	RELEFF
2	0.0	0	0.001	0.001
2	0.0	0	0.200	0.200
2	0.0	0	0.500	0.500
2	0.0	0	0.700	0.700
2	0.0	0	0.990	0.990
2	0.5	0	0.001	0.501
2	0.5	0	0.200	0.700
2	0.5	0	0.500	1.000
2	0.5	0	0.700	1.200
2	0.5	0	0.990	1.490
2	1.0	0	0.001	1.001
2	1.0	0	0.200	1.200
2	1.0	0	0.500	1.500
2	1.0	0	0.700	1.700
2	1.0	0	0.990	1.990
2	2.0	0	0.001	2.001
2	2.0	0	0.200	2.200

B	LAMBDA	RHO	GAMMA	RELEFF
2	2.0	0	0.500	2.500
2	2.0	0	0.700	2.700
2	2.0	0	0.990	2.990
2	5.0	0	0.001	5.001
2	5.0	0	0.200	5.200
2	5.0	0	0.500	5.500
2	5.0	0	0.700	5.700
2	5.0	0	0.990	5.990
3	0.0	0	0.001	0.001
3	0.0	0	0.200	0.200
3	0.0	0	0.500	0.500
3	0.0	0	0.700	0.700
3	0.0	0	0.990	0.990
3	0.5	0	0.001	0.501
3	0.5	0	0.200	0.700
3	0.5	0	0.500	1.000
3	0.5	0	0.700	1.200
3	0.5	0	0.990	1.490
3	1.0	0	0.001	1.001
3	1.0	0	0.200	1.200
3	1.0	0	0.500	1.500
3	1.0	0	0.700	1.700
3	1.0	0	0.990	1.990
3	2.0	0	0.001	2.001
3	2.0	0	0.200	2.200
3	2.0	0	0.500	2.500
3	2.0	0	0.700	2.700
3	2.0	0	0.990	2.990
3	5.0	0	0.001	5.001
3	5.0	0	0.200	5.200
3	5.0	0	0.500	5.500
3	5.0	0	0.700	5.700
3	5.0	0	0.990	5.990
6	0.0	0	0.001	0.001
6	0.0	0	0.200	0.200
6	0.0	0	0.500	0.500
6	0.0	0	0.700	0.700
6	0.0	0	0.990	0.990
6	0.5	0	0.001	0.501
6	0.5	0	0.200	0.700
6	0.5	0	0.500	1.000
6	0.5	0	0.700	1.200
6	0.5	0	0.990	1.490
6	1.0	0	0.001	1.001
6	1.0	0	0.200	1.200
6	1.0	0	0.500	1.500
6	1.0	0	0.700	1.700
6	1.0	0	0.990	1.990
6	2.0	0	0.001	2.001
6	2.0	0	0.200	2.200
6	2.0	0	0.500	2.500
6	2.0	0	0.700	2.700

B	LAMBDA	RHO	GAMMA	RELEFF
6	2.0	0	0.990	2.990
6	5.0	0	0.001	5.001
6	5.0	0	0.200	5.200
6	5.0	0	0.500	5.500
6	5.0	0	0.700	5.700
6	5.0	0	0.990	5.990
12	0.0	0	0.001	0.001
12	0.0	0	0.200	0.200
12	0.0	0	0.500	0.500
12	0.0	0	0.700	0.700
12	0.0	0	0.990	0.990
12	0.5	0	0.001	0.501
12	0.5	0	0.200	0.700
12	0.5	0	0.500	1.000
12	0.5	0	0.700	1.200
12	0.5	0	0.990	1.490
12	1.0	0	0.001	1.001
12	1.0	0	0.200	1.200
12	1.0	0	0.500	1.500
12	1.0	0	0.700	1.700
12	1.0	0	0.990	1.990
12	2.0	0	0.001	2.001
12	2.0	0	0.200	2.200
12	2.0	0	0.500	2.500
12	2.0	0	0.700	2.700
12	2.0	0	0.990	2.990
12	5.0	0	0.001	5.001
12	5.0	0	0.200	5.200
12	5.0	0	0.500	5.500
12	5.0	0	0.700	5.700
12	5.0	0	0.990	5.990

APPENDIX D
RELATIVE EFFICIENCIES FOR GIVEN PARAMETER VALUES OF γ, ρ, AND b FOR λ= 0

SAS Program

```
OPTIONS PS = 60;

DATA RELEFF;
   DO B = 2, 3, 6, 12;
      DO LAMBDA = 0;
         DO RHO = 0, .5, 1, 2, 5;
            DO GAMMA = .001, .2, .5, .7, .99;
               RELEFF =
               ((GAMMA*(1+B*RHO)*(1+B*RHO*GAMMA))+LAMBDA)/
                          (1+B*RHO*GAMMA)**2;
               OUTPUT;
            END;
         END;
      END;
   END;

PROC PRINT DATA = RELEFF NOOBS;
RUN;
```

Output

B	LAMBDA	RHO	GAMMA	RELEFF
2	0	0.0	0.001	0.00100
2	0	0.0	0.200	0.20000
2	0	0.0	0.500	0.50000
2	0	0.0	0.700	0.70000
2	0	0.0	0.990	0.99000
2	0	0.5	0.001	0.00200
2	0	0.5	0.200	0.33333
2	0	0.5	0.500	0.66667
2	0	0.5	0.700	0.82353
2	0	0.5	0.990	0.99497
2	0	1.0	0.001	0.00299
2	0	1.0	0.200	0.42857
2	0	1.0	0.500	0.75000
2	0	1.0	0.700	0.87500
2	0	1.0	0.990	0.99664
2	0	2.0	0.001	0.00498
2	0	2.0	0.200	0.55556

B	LAMBDA	RHO	GAMMA	RELEFF
2	0	2.0	0.500	0.83333
2	0	2.0	0.700	0.92105
2	0	2.0	0.990	0.99798
2	0	5.0	0.001	0.01089
2	0	5.0	0.200	0.73333
2	0	5.0	0.500	0.91667
2	0	5.0	0.700	0.96250
2	0	5.0	0.990	0.99908
3	0	0.0	0.001	0.00100
3	0	0.0	0.200	0.20000
3	0	0.0	0.500	0.50000
3	0	0.0	0.700	0.70000
3	0	0.0	0.990	0.99000
3	0	0.5	0.001	0.00250
3	0	0.5	0.200	0.38462
3	0	0.5	0.500	0.71429
3	0	0.5	0.700	0.85366
3	0	0.5	0.990	0.99598
3	0	1.0	0.001	0.00399
3	0	1.0	0.200	0.50000
3	0	1.0	0.500	0.80000
3	0	1.0	0.700	0.90323
3	0	1.0	0.990	0.99748
3	0	2.0	0.001	0.00696
3	0	2.0	0.200	0.63636
3	0	2.0	0.500	0.87500
3	0	2.0	0.700	0.94231
3	0	2.0	0.990	0.99856
3	0	5.0	0.001	0.01576
3	0	5.0	0.200	0.80000
3	0	5.0	0.500	0.94118
3	0	5.0	0.700	0.97391
3	0	5.0	0.990	0.99937
6	0	0.0	0.001	0.00100
6	0	0.0	0.200	0.20000
6	0	0.0	0.500	0.50000
6	0	0.0	0.700	0.70000
6	0	0.0	0.990	0.99000
6	0	0.5	0.001	0.00399
6	0	0.5	0.200	0.50000
6	0	0.5	0.500	0.80000
6	0	0.5	0.700	0.90323
6	0	0.5	0.990	0.99748
6	0	1.0	0.001	0.00696
6	0	1.0	0.200	0.63636
6	0	1.0	0.500	0.87500
6	0	1.0	0.700	0.94231
6	0	1.0	0.990	0.99856
6	0	2.0	0.001	0.01285
6	0	2.0	0.200	0.76471
6	0	2.0	0.500	0.92857
6	0	2.0	0.700	0.96809

B	LAMBDA	RHO	GAMMA	RELEFF
6	0	2.0	0.990	0.99922
6	0	5.0	0.001	0.03010
6	0	5.0	0.200	0.88571
6	0	5.0	0.500	0.96875
6	0	5.0	0.700	0.98636
6	0	5.0	0.990	0.99967
12	0	0.0	0.001	0.00100
12	0	0.0	0.200	0.20000
12	0	0.0	0.500	0.50000
12	0	0.0	0.700	0.70000
12	0	0.0	0.990	0.99000
12	0	0.5	0.001	0.00696
12	0	0.5	0.200	0.63636
12	0	0.5	0.500	0.87500
12	0	0.5	0.700	0.94231
12	0	0.5	0.990	0.99856
12	0	1.0	0.001	0.01285
12	0	1.0	0.200	0.76471
12	0	1.0	0.500	0.92857
12	0	1.0	0.700	0.96809
12	0	1.0	0.990	0.99922
12	0	2.0	0.001	0.02441
12	0	2.0	0.200	0.86207
12	0	2.0	0.500	0.96154
12	0	2.0	0.700	0.98315
12	0	2.0	0.990	0.99960
12	0	5.0	0.001	0.05755
12	0	5.0	0.200	0.93846
12	0	5.0	0.500	0.98387
12	0	5.0	0.700	0.99302
12	0	5.0	0.990	0.99983

APPENDIX E
SIMULATION PROGRAMS

Program 1 for a = 3

The following is an example of Program 1 for a = 3, b = 2, and γ = .001. The a, b, and γ value may be changed for the other parameter combinations utilizing Program 1 for a = 3.

```
      INTEGER A,B,N,CLASS,TIME,COUNT,SIZE,PASS,FC,COUNTER2,
         COUNT3,W
      INTEGER NUM,NUM2,COUNTER3
      REAL XMAT(2,3),XVEC(6),SUM,SUMCOL,XDOTI(3),XBAR,TSS2
      REAL WSS,TSS,GAMMA,SMAT(2,3),ZMAT(2,3),GAMVAL(5),
         LAMBDA
      REAL DELTA,DELTAN,DELTA0,CHANGE(3),NXBAR,EPS,X2SSQQ,
         X2PX1
      REAL X2VEC(3),X2MAT(2,3),X1CMAT(2,3),X2SUM,X1CLSMN(3)
      REAL SUM2,CLSSUM(3),Z,X2STD(3),X2SSQ,X2STDSQ,X2STDSUM
      REAL LAMVAL(5),ERVEC(3),EVEC(6),EMAT(2,3),SIGE,SIGR,
         BSS
      REAL X1STD(2,3),X1MSTD(3),BETA0,BETA1,BETA2,
         X1WISTD(2,3),BSS2
      REAL YB(3),YMAT(2,3),RHOVAL(5),RHOHAT,TSSSQ,
         RHOSQ(5),XWPXW
      REAL B1WHAT,B1BHAT,B1HAT1,B1STRHAT,SIGEHAT,
         SIGRHAT,YVEC(6)
      REAL RELEFF,X1PY,X1PR1Y,SUMY,YCLSSUM(3),YBAR,YPR2Y,
         B1HAT2
      REAL YCLSMN(3),XWPIXW,YPXW,YPXWY,YPBETY,YPXB,XBPXB,
         XBPIXB
      REAL YBET(3),YWI(2,3)
      A=3
      B=2
      N=A*B
C
      EPS = 0.00005
      GAMVAL(1) = .001
      GAMVAL(2) = .2
      GAMVAL(3) = .5
      GAMVAL(4) = .7
      GAMVAL(5) = .99
      LAMVAL(1) = 0
      LAMVAL(2) = .5
      LAMVAL(3) = 1
      LAMVAL(4) = 2
      LAMVAL(5) = 5
      RHOVAL(1) = 0
      RHOVAL(2) = .5
      RHOVAL(3) = 1
      RHOVAL(4) = 2
      RHOVAL(5) = 5
```

```
      RHOSQ(1) = SQRT(RHOVAL(1))
      RHOSQ(2) = SQRT(RHOVAL(2))
      RHOSQ(3) = SQRT(RHOVAL(3))
      RHOSQ(4) = SQRT(RHOVAL(4))
      RHOSQ(5) = SQRT(RHOVAL(5))
      X2VEC(1) = 0
      X2VEC(2) = 0
      X2VEC(3) = 0
      SIGE = 1
      BETA0 = 0
      BETA1 = 1
C
      WRITE(4,*) 'A',A,'B',B
      COUNT2 = 0
      COUNTER3 = 0
C
      CALL RNOPT(2)
      CALL RNSET(348)
C
      DO 860 M = 1,500
C
      DO 850 K = 1,1
C
      DO 840 L = 1,5
C
      DO 830 W = 1,5
C
      LOOPD = 0
    5 LOOPC = 0
      LOOPD = LOOPD + 1
      IF (LOOPD.LE.50) THEN
        GO TO 6
      ELSE
        WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
        GO TO 860
      ENDIF

      DO 250 J = 1,50
C
    6 RNUN(N,XVEC)
      DO 7 I = 1,N
         XVEC(I) = XVEC(I) * 15
    7 CONTINUE
C
      SUM = 0
      COUNT = 1
      DO 20 CLASS = 1,A
            SUMCOL = 0
            DO 10 TIME = 1,B
                XMAT(TIME,CLASS) = XVEC(COUNT)
                SUM = SUM + XMAT(TIME,CLASS)
                SUMCOL = SUMCOL + XMAT(TIME,CLASS)
                COUNT = COUNT + 1
   10       CONTINUE
```

```
            XDOTI(CLASS) = SUMCOL/B
   20 CONTINUE
C
      XBAR = SUM/N
      WSS = 0
      TSS = 0
C
      DO 40 CLASS = 1,A
         DO 30 TIME = 1,B
            SMAT(TIME,CLASS) = XMAT(TIME,CLASS) -
            XDOTI(CLASS)
            WSS = WSS +
            (SMAT(TIME,CLASS))*(SMAT(TIME,CLASS))
            TSS = TSS + (XMAT(TIME,CLASS) - XBAR)*
     +                          (XMAT(TIME,CLASS) - XBAR)
   30    CONTINUE
   40 CONTINUE
C
   50 GAMMA = WSS/TSS
      SUMNEW = 0
C
      IF ((GAMMA.GE.GAMVAL(K)-
         EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +   THEN
         GO TO 300
      ENDIF
C
      IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
         SIZE = 0
         FC = 0
         PASS = 1
      ENDIF
C
      IF (GAMMA.GT.GAMVAL(K)+EPS) THEN
         SIZE = 1
         FC = 0
         PASS = 1
      ENDIF
C
      DELTA = .5
      NXBAR = XBAR
C
   55 LOOPC = LOOPC + 1
      IF (LOOPC.LE.50) THEN
        GO TO 56
      ELSE
        GO TO 5
      ENDIF
   56 WSS = 0
      TSS = 0
      IF ((SIZE.EQ.0).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = .75
         ELSE
```

```
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 70 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 60 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
 60         CONTINUE
 70      CONTINUE
         DO 90 CLASS = 1,A
            DO 80 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))*
     +                     (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                     (ZMAT(TIME,CLASS) - NXBAR)
 80            CONTINUE
 90      CONTINUE
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 0
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
         GO TO 55
C
      ELSE IF ((SIZE.EQ.1).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = 1.5
         ELSE
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 + DELTA
         NXBAR = DELTAN*XBAR
         DO 110 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 100 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
 100        CONTINUE
 110     CONTINUE
         DO 130 CLASS = 1,A
            DO 120 TIME = 1,B
```

```
                WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
                TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
  120        CONTINUE
  130     CONTINUE
          GAMMA = WSS/TSS
          IF ((GAMMA.GE.GAMVAL(K)-
             EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +       THEN
             GO TO 300
          ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
             SIZE = 0
             FC = 1
          ELSE
             SIZE = 1
             FC = 0
          ENDIF
          GO TO 55
C
       ELSE IF ((SIZE.EQ.0).AND.(FC.EQ.1)) THEN
          DELTA0 = DELTAN
          DELTA = DELTA*(0.9)
          DELTAN = DELTA0 - DELTA
          NXBAR = DELTAN*XBAR
          DO 150 CLASS = 1,A
             CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
             DO 140 TIME = 1,B
                ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                   CHANGE(CLASS)
  140        CONTINUE
  150     CONTINUE
          DO 170 CLASS = 1,A
             DO 160 TIME = 1,B
                WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
                TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
  160        CONTINUE
  170     CONTINUE
          GAMMA = WSS/TSS
          IF ((GAMMA.GE.GAMVAL(K)-
             EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +       THEN
             GO TO 300
          ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
             SIZE = 0
             FC = 1
          ELSE
             SIZE = 1
```

```
            FC = 1
         ENDIF
         GO TO 55
C
      ELSE
         DELTA0 = DELTAN
         DELTA = DELTA*(0.9)
         DELTAN = DELTA0 + DELTA
         NXBAR = DELTAN*XBAR
         DO 190 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 180 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
  180       CONTINUE
  190    CONTINUE
         DO 210 CLASS = 1,A
            DO 200 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
  200       CONTINUE
  210    CONTINUE
C
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 1
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
         GO TO 55
C
      ENDIF
C
  250 CONTINUE
C
  300 IF ((GAMMA.GE.GAMVAL(K)-
           EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +     THEN
           GO TO 305
      ELSE
           WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
      ENDIF
C
  305 DO 320 CLASS = 1,A
```

```
            CLSSUM(CLASS) = 0
            DO 310 TIME = 1,B
               X1CMAT(TIME,CLASS) = ZMAT(TIME,CLASS) - NXBAR
               CLSSUM(CLASS) = CLSSUM(CLASS) +
                               X1CMAT(TIME,CLASS)
  310       CONTINUE
  320    CONTINUE
C
         CLSSUM1 = 0
         CLSSUM2 = 0
         CLSSUM3 = 0
         CLSSUM4 = 0
         CLSSUM5 = 0
         DO 322 CLASS = 1,A
            IF (CLASS.EQ.1) THEN
             CLSSUM1 = CLSSUM(CLASS)
            ELSE IF (CLASS.EQ.2) THEN
             CLSSUM2 = CLSSUM(CLASS)
            ELSE
             CLSSUM3 = CLSSUM(CLASS)
            ENDIF
  322     CONTINUE
C
         ZP = (CLSSUM1 - CLSSUM2)**2
         Z2 = (LAMVAL(L)*TSS*SIGE) / (ZP*GAMVAL(K))
         Z3 = SQRT(Z2)
         DO 325 CLASS = 1,A
           IF (CLASS.EQ.1) THEN
           X2VEC(CLASS) = Z3
           ELSE IF (CLASS.EQ.2) THEN
           X2VEC(CLASS) = -Z3
           ELSE
           X2VEC(CLASS) = 0
           ENDIF
  325    CONTINUE
C
          SUM2 = 0
          DO 390 CLASS = 1,A
             DO 385 TIME = 1,B
                X2MAT(TIME,CLASS) = X2VEC(CLASS)
  385        CONTINUE
  390     CONTINUE
C
          DO 396 CLASS = 1,A
             X1CLSMN(CLASS) = CLSSUM(CLASS)/B
  396     CONTINUE
C
          BSS = 0
          DO 402 CLASS = 1,A
             DO 401 TIME = 1,B
                BSS = BSS + (X1CLSMN(CLASS)*X1CLSMN(CLASS))
  401        CONTINUE
  402     CONTINUE
C
```

```
      X2PX1 = 0
      DO 404 CLASS = 1,A
         DO 403 TIME = 1,B
            X2PX1 = X2PX1
     +          + (X2MAT(TIME,CLASS)*X1CMAT(TIME,CLASS))
 403     CONTINUE
 404  CONTINUE
C
      Z = (SIGE*TSS*TSS*LAMVAL(L))/((X2PX1*X2PX1)*BSS)
      BETA2 = SQRT(Z)
C
      LAMBDA = (BETA2*X2PX1*X2PX1*BETA2*BSS)/(SIGE*TSS*TSS)
C
      X2SSQ = 0
      X2STDSQ = 0
      VARX2STD = 0
      DO 410 CLASS = 1,A
         X2SSQ = X2SSQ + (B*(X2VEC(CLASS)**2))
 410  CONTINUE
C
      X2SSQQ = SQRT(X2SSQ)
      DO 415 CLASS = 1,A
         X2STD(CLASS) = X2VEC(CLASS)/X2SSQQ
         X2STDSQ = X2STDSQ + (B*(X2STD(CLASS)**2))
         VARX2STD = X2STDSQ / (N-1)
 415  CONTINUE
C
      IF (VARX2STD.GT.1.5) THEN
         COUNTER2 = COUNTER2 + 1
      ELSE
         COUNTER2 = COUNTER2
      ENDIF
C
     CALL RNNOA(A,ERVEC)
     CALL RNNOA(N,EVEC)
     COUNT3 = 1
     DO 435 CLASS = 1,A
        DO 434 TIME = 1,B
           EMAT(TIME,CLASS) = EVEC(COUNT3)
           COUNT3 = COUNT3 + 1
 434    CONTINUE
 435 CONTINUE
 440 CONTINUE
C
     TSSSQ = SQRT(TSS)
     DO 450 CLASS = 1,A
        X1MSTD(CLASS) = (CLSSUM(CLASS)/B)/TSSSQ
 450 CONTINUE
C
     DO 470 CLASS = 1,A
        DO 460 TIME = 1,B
           X1STD(TIME,CLASS) = X1CMAT(TIME,CLASS)/TSSSQ
 460    CONTINUE
 470 CONTINUE
```

```
C
      DO 510 CLASS = 1,A
         YB(CLASS) = BETA0 + (BETA1*X1MSTD(CLASS)) +
     +                (BETA2*X2STD(CLASS)) +
     +                (RHOSQ(W)*ERVEC(CLASS))
  510 CONTINUE
C
      DO 540 CLASS = 1,A
         DO 530 TIME = 1,B
            X1WISTD(TIME,CLASS) = X1STD(TIME,CLASS) -
                                  X1MSTD(CLASS)
  530    CONTINUE
  540 CONTINUE
C
      DO 580 CLASS = 1,A
         DO 570 TIME = 1,B
            YMAT(TIME,CLASS) = YB(CLASS) +
            (BETA1*X1WISTD(TIME,CLASS))  SIM03720
     +                             + EMAT(TIME,CLASS)
  570    CONTINUE
  580 CONTINUE
C
      X1PY = 0
      DO 620 CLASS = 1,A
         DO 610 TIME = 1,B
            X1PY = X1PY +
               (X1STD(TIME,CLASS)*YMAT(TIME,CLASS))
  610 CONTINUE
  620 CONTINUE
C
      B1HAT1 = X1PY
C
      X1PR1Y = 0
      DO 640 CLASS = 1,A
         DO 630 TIME = 1,B
            X1PR1Y = X1PR1Y +
                    (X1MSTD(CLASS)*YMAT(TIME,CLASS))
  630 CONTINUE
  640 CONTINUE
C
      B1BHAT = X1PR1Y/(BSS/TSS)
C
      X1PR2Y = 0
      DO 660 CLASS = 1,A
         DO 650 TIME = 1,B
            X1PR2Y = X1PR2Y +
                 (X1WISTD(TIME,CLASS)*YMAT(TIME,CLASS))
  650 CONTINUE
  660 CONTINUE
C
      B1WHAT = X1PR2Y/(WSS/TSS)
C
      SUMY = 0
      DO 680 CLASS = 1,A
```

```
          YCLSSUM(CLASS) = 0
          DO 670 TIME = 1,B
             SUMY = SUMY + YMAT(TIME,CLASS)
             YCLSSUM(CLASS) = YCLSSUM(CLASS) +
                YMAT(TIME,CLASS)
  670     CONTINUE
  680  CONTINUE
C
       DO 685 CLASS = 1,A
          YCLSMN(CLASS) = YCLSSUM(CLASS)/B
  685  CONTINUE
C
       YBAR = SUMY/(A*B)
       YPR2Y = 0
       DO 700 CLASS = 1,A
          DO 690 TIME = 1,B
             YWI(TIME,CLASS) = YMAT(TIME,CLASS) -
                YCLSMN(CLASS)
             YPR2Y = YPR2Y + YWI(TIME,CLASS)*YWI(TIME,CLASS))
  690     CONTINUE
  700  CONTINUE
C
       XWPXW = 0
       YPXW = 0
       DO 720 CLASS = 1,A
          DO 710 TIME = 1,B
             YPXW = YPXW +
                       (YMAT(TIME,CLASS)*X1WISTD(TIME,CLASS))
             XWPXW = XWPXW +
                    (X1WISTD(TIME,CLASS)*X1WISTD(TIME,CLASS))
             XWPIXW = 1/XWPXW
  710     CONTINUE
  720  CONTINUE
C
       YPXWY = (YPXW*YPXW*XWPIXW)
C
       SIGEHAT = (YPR2Y - YPXWY)/(A*(B-1)-1)
C
       YPBETY = 0
       DO 750 CLASS = 1,A
          YBET(CLASS) = YCLSMN(CLASS) - YBAR
          YPBETY = YPBETY + B*(YBET(CLASS)*YBET(CLASS))
  750  CONTINUE
C
       YPXB = 0
       XBPXB = 0
       DO 770 CLASS = 1,A
          DO 760 TIME = 1,B
             YPXB = YPXB + (YMAT(TIME,CLASS)*X1MSTD(CLASS))
             XBPXB = XBPXB + (X1MSTD(CLASS)*X1MSTD(CLASS))
             XBPIXB = 1/XBPXB
  760     CONTINUE
  770  CONTINUE
C
```

```
      YPXBY = YPXB*YPXB*XBPIXB
C
      SIGRHAT = (((YPBETY - YPXBY)/(A-2)) - SIGEHAT)/B
      COUNTER3 = 0
      IF (SIGRHAT.LT.0) THEN
         RHOHAT = 0
         COUNTER3 = COUNTER3 + 1
      ELSE
         RHOHAT = SIGRHAT/SIGEHAT
      ENDIF
C
      D1 = BSS/TSS
      D2 = WSS/TSS
      D3 = 1/((B*SIGRHAT) + SIGEHAT)
      D4 = 1/SIGEHAT
      B1HAT2 = ((D1*B1BHAT) + (D2*B1WHAT))/(D1 + D2)
      B1STRHAT = ((D1*D3*B1BHAT) + (D2*D4*B1WHAT))/((D1*D3)
                 + (D2*D4))
C
      RELEFF = (((1+(B*RHOHAT*GAMVAL(K)))*
                (1+(B*RHOHAT))*GAMVAL(K))
     +           + LAMVAL(L))/((1+(B*RHOHAT*GAMVAL(K)))**2)
C
      WRITE(4,*) LAMVAL(L),RHOVAL(W),RELEFF
  830 CONTINUE
C
  840 CONTINUE
C
  850 CONTINUE
C
  860 CONTINUE
      WRITE(4,*) 'CVARX2',COUNTER2,'CRNEG',COUNTER3
C
  900 END
```

Program 1 for a = 5

The following is an example of Program 1 for a = 5, b = 2, and γ = .001. The a, b, and γ value may be changed for the other parameter combinations utilizing Program 1 for a = 5.

```
      INTEGER A,B,N,CLASS,TIME,COUNT,SIZE,PASS,FC,COUNTER2,
         COUNT3,W
      INTEGER NUM,NUM2,COUNTER3
      REAL XMAT(2,5),XVEC(10),SUM,SUMCOL,XDOTI(5),XBAR,TSS2
      REAL WSS,TSS,GAMMA,SMAT(2,5),ZMAT(2,5),GAMVAL(5),
         LAMBDA
      REAL DELTA,DELTAN,DELTA0,CHANGE(5),NXBAR,EPS,X2SSQQ,
         X2PX1
      REAL X2VEC(5),X2MAT(2,5),X1CMAT(2,5),X2SUM,X1CLSMN(3)
      REAL SUM2,CLSSUM(5),Z,X2STD(5),X2SSQ,X2STDSQ,X2STDSUM
      REAL LAMVAL(5),ERVEC(5),EVEC(10),EMAT(2,5),SIGE,SIGR,
```

```
      BSS
      REAL X1STD(2,5),X1MSTD(5),BETA0,BETA1,BETA2,
         X1WISTD(2,5),BSS2
      REAL YB(5),YMAT(2,5),RHOVAL(5),RHOHAT,TSSSQ,
         RHOSQ(5),XWPXW
      REAL B1WHAT,B1BHAT,B1HAT1,B1STRHAT,SIGEHAT,
         SIGRHAT,YVEC(10)
      REAL RELEFF,X1PY,X1PR1Y,SUMY,YCLSSUM(5),YBAR,YPR2Y,
         B1HAT2
      REAL YCLSMN(5),XWPIXW,YPXW,YPXWY,YPBETY,YPXB,XBPXB,
         XBPIXB
      REAL YBET(5),YWI(2,5)
      A=5
      B=2
      N=A*B
C
      EPS = 0.00005
      GAMVAL(1) = .001
      GAMVAL(2) = .2
      GAMVAL(3) = .5
      GAMVAL(4) = .7
      GAMVAL(5) = .99
      LAMVAL(1) = 0
      LAMVAL(2) = .5
      LAMVAL(3) = 1
      LAMVAL(4) = 2
      LAMVAL(5) = 5
      RHOVAL(1) = 0
      RHOVAL(2) = .5
      RHOVAL(3) = 1
      RHOVAL(4) = 2
      RHOVAL(5) = 5
      RHOSQ(1) = SQRT(RHOVAL(1))
      RHOSQ(2) = SQRT(RHOVAL(2))
      RHOSQ(3) = SQRT(RHOVAL(3))
      RHOSQ(4) = SQRT(RHOVAL(4))
      RHOSQ(5) = SQRT(RHOVAL(5))
      X2VEC(1) = 0
      X2VEC(2) = 0
      X2VEC(3) = 0
      X2VEC(4) = 0
      X2VEC(5) = 0
      SIGE = 1
      BETA0 = 0
      BETA1 = 1
C
      WRITE(4,*) 'A',A,'B',B
      COUNT2 = 0
      COUNTER3 = 0
C
      CALL RNOPT(2)
      CALL RNSET(162512)
C
      DO 860 M = 1,1000
```

```
C
      DO 850 K = 1,1
C
      DO 840 L = 1,5
C
      DO 830 W = 1,5
C
      LOOPD = 0
    5 LOOPC = 0
      LOOPD = LOOPD + 1
      IF (LOOPD.LE.50) THEN
        GO TO 6
      ELSE
        WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
        GO TO 860
      ENDIF

      DO 250 J = 1,50
C
    6 RNUN(N,XVEC)
      DO 7 I = 1,N
         XVEC(I) = XVEC(I) * 15
    7 CONTINUE
C
      SUM = 0
      COUNT = 1
      DO 20 CLASS = 1,A
            SUMCOL = 0
            DO 10 TIME = 1,B
                XMAT(TIME,CLASS) = XVEC(COUNT)
                SUM = SUM + XMAT(TIME,CLASS)
                SUMCOL = SUMCOL + XMAT(TIME,CLASS)
                COUNT = COUNT + 1
   10       CONTINUE
            XDOTI(CLASS) = SUMCOL/B
   20 CONTINUE
C
      XBAR = SUM/N
      WSS = 0
      TSS = 0
C
      DO 40 CLASS = 1,A
         DO 30 TIME = 1,B
            SMAT(TIME,CLASS) = XMAT(TIME,CLASS) -
            XDOTI(CLASS)
            WSS = WSS +
            (SMAT(TIME,CLASS))*(SMAT(TIME,CLASS))
            TSS = TSS + (XMAT(TIME,CLASS) - XBAR)*
     +                            (XMAT(TIME,CLASS) - XBAR)
   30    CONTINUE
   40 CONTINUE
C
   50 GAMMA = WSS/TSS
      SUMNEW = 0
```

```
C
      IF ((GAMMA.GE.GAMVAL(K)-
         EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +   THEN
         GO TO 300
      ENDIF
C
      IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
         SIZE = 0
         FC = 0
         PASS = 1
      ENDIF
C
      IF (GAMMA.GT.GAMVAL(K)+EPS) THEN
         SIZE = 1
         FC = 0
         PASS = 1
      ENDIF
C
      DELTA = .5
      NXBAR = XBAR
C
   55 LOOPC = LOOPC + 1
      IF (LOOPC.LE.50) THEN
        GO TO 56
      ELSE
        GO TO 5
      ENDIF
   56 WSS = 0
      TSS = 0
      IF ((SIZE.EQ.0).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = .75
         ELSE
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 70 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 60 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
   60       CONTINUE
   70    CONTINUE
         DO 90 CLASS = 1,A
            DO 80 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))*
     +                     (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                     (ZMAT(TIME,CLASS) - NXBAR)
```

```
 80           CONTINUE
 90       CONTINUE
          GAMMA = WSS/TSS
          IF ((GAMMA.GE.GAMVAL(K)-
             EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +       THEN
             GO TO 300
          ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
             SIZE = 0
             FC = 0
          ELSE
             SIZE = 1
             FC = 1
          ENDIF
          GO TO 55
C
       ELSE IF ((SIZE.EQ.1).AND.(FC.EQ.0)) THEN
          IF (PASS.EQ.1) THEN
             DELTA0 = 1.5
          ELSE
             DELTA0 = DELTAN
          ENDIF
          PASS = 0
          DELTAN = DELTA0 + DELTA
          NXBAR = DELTAN*XBAR
          DO 110 CLASS = 1,A
             CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
             DO 100 TIME = 1,B
                ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                   CHANGE(CLASS)
100          CONTINUE
110       CONTINUE
          DO 130 CLASS = 1,A
             DO 120 TIME = 1,B
                WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
                TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
120          CONTINUE
130       CONTINUE
          GAMMA = WSS/TSS
          IF ((GAMMA.GE.GAMVAL(K)-
             EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +       THEN
             GO TO 300
          ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
             SIZE = 0
             FC = 1
          ELSE
             SIZE = 1
             FC = 0
          ENDIF
```

```
         GO TO 55
C
      ELSE IF ((SIZE.EQ.0).AND.(FC.EQ.1)) THEN
         DELTA0 = DELTAN
         DELTA = DELTA*(0.9)
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 150 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 140 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
 140        CONTINUE
 150     CONTINUE
         DO 170 CLASS = 1,A
            DO 160 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
 160        CONTINUE
 170     CONTINUE
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 1
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
         GO TO 55
C
      ELSE
         DELTA0 = DELTAN
         DELTA = DELTA*(0.9)
         DELTAN = DELTA0 + DELTA
         NXBAR = DELTAN*XBAR
         DO 190 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 180 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
 180        CONTINUE
 190     CONTINUE
         DO 210 CLASS = 1,A
            DO 200 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
```

```
     +                          (ZMAT(TIME,CLASS) -
                                 CHANGE(CLASS))
                 TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                          (ZMAT(TIME,CLASS) - NXBAR)
  200          CONTINUE
  210       CONTINUE
C
            GAMMA = WSS/TSS
            IF ((GAMMA.GE.GAMVAL(K)-
               EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +         THEN
               GO TO 300
            ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
               SIZE = 0
               FC = 1
            ELSE
               SIZE = 1
               FC = 1
            ENDIF
            GO TO 55
C
         ENDIF
C
  250 CONTINUE
C
  300 IF ((GAMMA.GE.GAMVAL(K)-
              EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +        THEN
              GO TO 305
      ELSE
              WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
      ENDIF
C
  305 DO 320 CLASS = 1,A
            CLSSUM(CLASS) = 0
            DO 310 TIME = 1,B
               X1CMAT(TIME,CLASS) = ZMAT(TIME,CLASS) - NXBAR
               CLSSUM(CLASS) = CLSSUM(CLASS) +
                               X1CMAT(TIME,CLASS)
  310       CONTINUE
  320 CONTINUE
C
      CLSSUM1 = 0
      CLSSUM2 = 0
      CLSSUM3 = 0
      CLSSUM4 = 0
      CLSSUM5 = 0
      DO 322 CLASS = 1,A
            IF (CLASS.EQ.1) THEN
             CLSSUM1 = CLSSUM(CLASS)
            ELSE IF (CLASS.EQ.2) THEN
             CLSSUM2 = CLSSUM(CLASS)
             ELSE IF (CLASS.EQ.3) THEN
             CLSSUM3 = CLSSUM(CLASS)
```

```
          ELSE IF (CLASS.EQ.4) THEN
          CLSSUM4 = CLSSUM(CLASS)
         ELSE
          CLSSUM5 = CLSSUM(CLASS)
         ENDIF
  322  CONTINUE
C
      ZP = (CLSSUM1 - CLSSUM2)**2
      Z2 = (LAMVAL(L)*TSS*SIGE) / (ZP*GAMVAL(K))
      Z3 = SQRT(Z2)
      DO 325 CLASS = 1,A
        IF (CLASS.EQ.1) THEN
        X2VEC(CLASS) = Z3
        ELSE IF (CLASS.EQ.2) THEN
        X2VEC(CLASS) = -Z3
        ELSE IF (CLASS.EQ.3) THEN
        X2VEC(CLASS) = Z3/2
        ELSE IF (CLASS.EQ.4) THEN
        X2VEC(CLASS) = -Z3/2
        ELSE
        X2VEC(CLASS) = 0
        ENDIF
  325 CONTINUE
C
       SUM2 = 0
       DO 390 CLASS = 1,A
          DO 385 TIME = 1,B
             X2MAT(TIME,CLASS) = X2VEC(CLASS)
  385     CONTINUE
  390  CONTINUE
C
       DO 396 CLASS = 1,A
          X1CLSMN(CLASS) = CLSSUM(CLASS)/B
  396  CONTINUE
C
       BSS = 0
       DO 402 CLASS = 1,A
          DO 401 TIME = 1,B
             BSS = BSS + (X1CLSMN(CLASS)*X1CLSMN(CLASS))
  401     CONTINUE
  402  CONTINUE
C
       X2PX1 = 0
       DO 404 CLASS = 1,A
          DO 403 TIME = 1,B
             X2PX1 = X2PX1
                 + (X2MAT(TIME,CLASS)*X1CMAT(TIME,CLASS))
  403     CONTINUE
  404  CONTINUE
C
       Z = (SIGE*TSS*TSS*LAMVAL(L))/((X2PX1*X2PX1)*BSS)
       BETA2 = SQRT(Z)
C
       LAMBDA = (BETA2*X2PX1*X2PX1*BETA2*BSS)/(SIGE*TSS*TSS)
```

```
C
       X2SSQ = 0
       X2STDSQ = 0
       VARX2STD = 0
       DO 410 CLASS = 1,A
          X2SSQ = X2SSQ + (B*(X2VEC(CLASS)**2))
 410   CONTINUE
C
       X2SSQQ = SQRT(X2SSQ)
       DO 415 CLASS = 1,A
          X2STD(CLASS) = X2VEC(CLASS)/X2SSQQ
          X2STDSQ = X2STDSQ + (B*(X2STD(CLASS)**2))
          VARX2STD = X2STDSQ / (N-1)
 415   CONTINUE
C
       IF (VARX2STD.GT.1.5) THEN
          COUNTER2 = COUNTER2 + 1
       ELSE
          COUNTER2 = COUNTER2
       ENDIF
C
      CALL RNNOA(A,ERVEC)
      CALL RNNOA(N,EVEC)
      COUNT3 = 1
      DO 435 CLASS = 1,A
         DO 434 TIME = 1,B
            EMAT(TIME,CLASS) = EVEC(COUNT3)
            COUNT3 = COUNT3 + 1
 434     CONTINUE
 435  CONTINUE
 440  CONTINUE
C
      TSSSQ = SQRT(TSS)
      DO 450 CLASS = 1,A
         X1MSTD(CLASS) = (CLSSUM(CLASS)/B)/TSSSQ
 450  CONTINUE
C
      DO 470 CLASS = 1,A
         DO 460 TIME = 1,B
            X1STD(TIME,CLASS) = X1CMAT(TIME,CLASS)/TSSSQ
 460     CONTINUE
 470  CONTINUE
C
      DO 510 CLASS = 1,A
         YB(CLASS) = BETA0 + (BETA1*X1MSTD(CLASS)) +
     +                (BETA2*X2STD(CLASS)) +
     +                (RHOSQ(W)*ERVEC(CLASS))
 510  CONTINUE
C
      DO 540 CLASS = 1,A
         DO 530 TIME = 1,B
            X1WISTD(TIME,CLASS) = X1STD(TIME,CLASS) -
                                  X1MSTD(CLASS)
 530     CONTINUE
```

```
  540 CONTINUE
C
      DO 580 CLASS = 1,A
         DO 570 TIME = 1,B
            YMAT(TIME,CLASS) = YB(CLASS) +
            (BETA1*X1WISTD(TIME,CLASS))   SIM03720
     +                            + EMAT(TIME,CLASS)
  570    CONTINUE
  580 CONTINUE
C
      X1PY = 0
      DO 620 CLASS = 1,A
         DO 610 TIME = 1,B
            X1PY = X1PY +
               (X1STD(TIME,CLASS)*YMAT(TIME,CLASS))
  610 CONTINUE
  620 CONTINUE
C
      B1HAT1 = X1PY
C
      X1PR1Y = 0
      DO 640 CLASS = 1,A
         DO 630 TIME = 1,B
            X1PR1Y = X1PR1Y +
                        (X1MSTD(CLASS)*YMAT(TIME,CLASS))
  630 CONTINUE
  640 CONTINUE
C
      B1BHAT = X1PR1Y/(BSS/TSS)
C
      X1PR2Y = 0
      DO 660 CLASS = 1,A
         DO 650 TIME = 1,B
            X1PR2Y = X1PR2Y +
                     (X1WISTD(TIME,CLASS)*YMAT(TIME,CLASS))
  650 CONTINUE
  660 CONTINUE
C
      B1WHAT = X1PR2Y/(WSS/TSS)
C
      SUMY = 0
      DO 680 CLASS = 1,A
         YCLSSUM(CLASS) = 0
         DO 670 TIME = 1,B
            SUMY = SUMY + YMAT(TIME,CLASS)
            YCLSSUM(CLASS) = YCLSSUM(CLASS) +
               YMAT(TIME,CLASS)
  670    CONTINUE
  680 CONTINUE
C
      DO 685 CLASS = 1,A
         YCLSMN(CLASS) = YCLSSUM(CLASS)/B
  685 CONTINUE
C
```

```
      YBAR = SUMY/(A*B)
      YPR2Y = 0
      DO 700 CLASS = 1,A
         DO 690 TIME = 1,B
            YWI(TIME,CLASS) = YMAT(TIME,CLASS) -
               YCLSMN(CLASS)
            YPR2Y = YPR2Y + YWI(TIME,CLASS)*YWI(TIME,CLASS))
  690    CONTINUE
  700 CONTINUE
C
      XWPXW = 0
      YPXW = 0
      DO 720 CLASS = 1,A
         DO 710 TIME = 1,B
            YPXW = YPXW +
                       (YMAT(TIME,CLASS)*X1WISTD(TIME,CLASS))
            XWPXW = XWPXW +
                    (X1WISTD(TIME,CLASS)*X1WISTD(TIME,CLASS))
            XWPIXW = 1/XWPXW
  710    CONTINUE
  720 CONTINUE
C
      YPXWY = (YPXW*YPXW*XWPIXW)
C
      SIGEHAT = (YPR2Y - YPXWY)/(A*(B-1)-1)
C
      YPBETY = 0
      DO 750 CLASS = 1,A
         YBET(CLASS) = YCLSMN(CLASS) - YBAR
         YPBETY = YPBETY + B*(YBET(CLASS)*YBET(CLASS))
  750 CONTINUE
C
      YPXB = 0
      XBPXB = 0
      DO 770 CLASS = 1,A
         DO 760 TIME = 1,B
            YPXB = YPXB + (YMAT(TIME,CLASS)*X1MSTD(CLASS))
            XBPXB = XBPXB + (X1MSTD(CLASS)*X1MSTD(CLASS))
            XBPIXB = 1/XBPXB
  760    CONTINUE
  770 CONTINUE
C
      YPXBY = YPXB*YPXB*XBPIXB
C
      SIGRHAT = (((YPBETY - YPXBY)/(A-2)) - SIGEHAT)/B
      COUNTER3 = 0
      IF (SIGRHAT.LT.0) THEN
         RHOHAT = 0
         COUNTER3 = COUNTER3 + 1
      ELSE
         RHOHAT = SIGRHAT/SIGEHAT
      ENDIF
C
      D1 = BSS/TSS
```

```
      D2 = WSS/TSS
      D3 = 1/((B*SIGRHAT) + SIGEHAT)
      D4 = 1/SIGEHAT
      B1HAT2 = ((D1*B1BHAT) + (D2*B1WHAT))/(D1 + D2)
      B1STRHAT = ((D1*D3*B1BHAT) + (D2*D4*B1WHAT))/((D1*D3)
                 + (D2*D4))
C
      RELEFF = (((1+(B*RHOHAT*GAMVAL(K)))*
                (1+(B*RHOHAT))*GAMVAL(K))
     +           + LAMVAL(L))/((1+(B*RHOHAT*GAMVAL(K)))**2)
C
      WRITE(4,*) LAMVAL(L),RHOVAL(W),RELEFF
  830 CONTINUE
C
  840 CONTINUE
C
  850 CONTINUE
C
  860 CONTINUE
      WRITE(4,*) 'CVARX2',COUNTER2,'CRNEG',COUNTER3
C
  900 END
```

Program 2 for a = 3

The following is an example of Program 2 for a = 3, b = 2, and γ = .2. The a, b, and γ value may be changed for the other parameter combinations utilizing Program 2 for a = 3.

```
      INTEGER A,B,N,CLASS,TIME,COUNT,SIZE,PASS,FC,COUNTER2,
         COUNT3,W
      INTEGER NUM,NUM2,COUNTER3
      REAL XMAT(2,3),XVEC(6),SUM,SUMCOL,XDOTI(3),XBAR,TSS2
      REAL WSS,TSS,GAMMA,SMAT(2,3),ZMAT(2,3),GAMVAL(5),
         LAMBDA
      REAL DELTA,DELTAN,DELTA0,CHANGE(3),NXBAR,EPS,X2SSQQ,
         X2PX1
      REAL X2VEC(3),X2MAT(2,3),X1CMAT(2,3),X2SUM,X1CLSMN(3)
      REAL SUM2,CLSSUM(3),Z,X2STD(3),X2SSQ,X2STDSQ,X2STDSUM
      REAL LAMVAL(5),ERVEC(3),EVEC(6),EMAT(2,3),SIGE,SIGR,
         BSS
      REAL X1STD(2,3),X1MSTD(3),BETA0,BETA1,BETA2,
         X1WISTD(2,3),BSS2
      REAL YB(3),YMAT(2,3),RHOVAL(5),RHOHAT,TSSSQ,
         RHOSQ(5),XWPXW
      REAL B1WHAT,B1BHAT,B1HAT1,B1STRHAT,SIGEHAT,
         SIGRHAT,YVEC(6)
      REAL RELEFF,X1PY,X1PR1Y,SUMY,YCLSSUM(3),YBAR,YPR2Y,
         B1HAT2
      REAL YCLSMN(3),XWPIXW,YPXW,YPXWY,YPBETY,YPXB,XBPXB,
         XBPIXB
      REAL YBET(3),YWI(2,3)
```

```
      A=3
      B=2
      N=A*B
C
      EPS = 0.00005
      GAMVAL(1) = .001
      GAMVAL(2) = .2
      GAMVAL(3) = .5
      GAMVAL(4) = .7
      GAMVAL(5) = .99
      LAMVAL(1) = 0
      LAMVAL(2) = .5
      LAMVAL(3) = 1
      LAMVAL(4) = 2
      LAMVAL(5) = 5
      RHOVAL(1) = 0
      RHOVAL(2) = .5
      RHOVAL(3) = 1
      RHOVAL(4) = 2
      RHOVAL(5) = 5
      RHOSQ(1) = SQRT(RHOVAL(1))
      RHOSQ(2) = SQRT(RHOVAL(2))
      RHOSQ(3) = SQRT(RHOVAL(3))
      RHOSQ(4) = SQRT(RHOVAL(4))
      RHOSQ(5) = SQRT(RHOVAL(5))
      X2VEC(1) = 0
      X2VEC(2) = 0
      X2VEC(3) = 0
      SIGE = 1
      BETA0 = 0
      BETA1 = 1
C
      WRITE(4,*) 'A',A,'B',B
      COUNT2 = 0
      COUNTER3 = 0
C
      CALL RNOPT(2)
      CALL RNSET(265)
C
      DO 860 M = 1,500
C
      DO 850 K = 2,2
C
      DO 840 L = 1,5
C
      DO 830 W = 1,5
C
      LOOPD = 0
    5 LOOPC = 0
      LOOPD = LOOPD + 1
      IF (LOOPD.LE.50) THEN
        GO TO 6
      ELSE
        WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
```

```
         GO TO 860
       ENDIF

       DO 250 J = 1,50
C
     6 RNUN(N,XVEC)
       DO 7 I = 1,N
          XVEC(I) = XVEC(I) * 15
     7 CONTINUE
C
       SUM = 0
       COUNT = 1
       DO 20 CLASS = 1,A
             SUMCOL = 0
             DO 10 TIME = 1,B
                 XMAT(TIME,CLASS) = XVEC(COUNT)
                 SUM = SUM + XMAT(TIME,CLASS)
                 SUMCOL = SUMCOL + XMAT(TIME,CLASS)
                 COUNT = COUNT + 1
   10        CONTINUE
             XDOTI(CLASS) = SUMCOL/B
   20 CONTINUE
C
       XBAR = SUM/N
       WSS = 0
       TSS = 0
C
       DO 40 CLASS = 1,A
          DO 30 TIME = 1,B
             SMAT(TIME,CLASS) = XMAT(TIME,CLASS) -
             XDOTI(CLASS)
             WSS = WSS +
             (SMAT(TIME,CLASS))*(SMAT(TIME,CLASS))
             TSS = TSS + (XMAT(TIME,CLASS) - XBAR)*
     +                              (XMAT(TIME,CLASS) - XBAR)
   30     CONTINUE
   40 CONTINUE
C
   50 GAMMA = WSS/TSS
       SUMNEW = 0
C
       IF ((GAMMA.GE.GAMVAL(K)-
          EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +    THEN
          GO TO 300
       ENDIF
C
       IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
          SIZE = 0
          FC = 0
          PASS = 1
       ENDIF
C
       IF (GAMMA.GT.GAMVAL(K)+EPS) THEN
```

```
         SIZE = 1
         FC = 0
         PASS = 1
      ENDIF
C
      DELTA = .5
      NXBAR = XBAR
C
   55 LOOPC = LOOPC + 1
      IF (LOOPC.LE.50) THEN
        GO TO 56
      ELSE
        GO TO 5
      ENDIF
   56 WSS = 0
      TSS = 0
      IF ((SIZE.EQ.0).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = .75
         ELSE
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 70 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 60 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
   60       CONTINUE
   70    CONTINUE
         DO 90 CLASS = 1,A
            DO 80 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
   80          CONTINUE
   90    CONTINUE
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 0
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
```

```
         GO TO 55
C
      ELSE IF ((SIZE.EQ.1).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = 1.5
         ELSE
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 + DELTA
         NXBAR = DELTAN*XBAR
         DO 110 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 100 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
 100        CONTINUE
 110     CONTINUE
         DO 130 CLASS = 1,A
            DO 120 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
 120        CONTINUE
 130     CONTINUE
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 1
         ELSE
            SIZE = 1
            FC = 0
         ENDIF
         GO TO 55
C
      ELSE IF ((SIZE.EQ.0).AND.(FC.EQ.1)) THEN
         DELTA0 = DELTAN
         DELTA = DELTA*(0.7)
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 150 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 140 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
 140        CONTINUE
 150     CONTINUE
```

```
          DO 170 CLASS = 1,A
             DO 160 TIME = 1,B
                WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
                TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
 160         CONTINUE
 170      CONTINUE
          GAMMA = WSS/TSS
          IF ((GAMMA.GE.GAMVAL(K)-
             EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +       THEN
             GO TO 300
          ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
             SIZE = 0
             FC = 1
          ELSE
             SIZE = 1
             FC = 1
          ENDIF
          GO TO 55
C
       ELSE
          DELTA0 = DELTAN
          DELTA = DELTA*(0.7)
          DELTAN = DELTA0 + DELTA
          NXBAR = DELTAN*XBAR
          DO 190 CLASS = 1,A
             CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
             DO 180 TIME = 1,B
                ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                   CHANGE(CLASS)
 180         CONTINUE
 190      CONTINUE
          DO 210 CLASS = 1,A
             DO 200 TIME = 1,B
                WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
                TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
 200         CONTINUE
 210      CONTINUE
C
          GAMMA = WSS/TSS
          IF ((GAMMA.GE.GAMVAL(K)-
             EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +       THEN
             GO TO 300
          ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
             SIZE = 0
```

```
            FC = 1
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
         GO TO 55
C
      ENDIF
C
  250 CONTINUE
C
  300 IF ((GAMMA.GE.GAMVAL(K)-
          EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +    THEN
          GO TO 305
      ELSE
          WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
      ENDIF
C
  305 DO 320 CLASS = 1,A
         CLSSUM(CLASS) = 0
         DO 310 TIME = 1,B
            X1CMAT(TIME,CLASS) = ZMAT(TIME,CLASS) - NXBAR
            CLSSUM(CLASS) = CLSSUM(CLASS) +
                               X1CMAT(TIME,CLASS)
  310    CONTINUE
  320 CONTINUE
C
      CLSSUM1 = 0
      CLSSUM2 = 0
      CLSSUM3 = 0
      CLSSUM4 = 0
      CLSSUM5 = 0
      DO 322 CLASS = 1,A
         IF (CLASS.EQ.1) THEN
          CLSSUM1 = CLSSUM(CLASS)
         ELSE IF (CLASS.EQ.2) THEN
          CLSSUM2 = CLSSUM(CLASS)
         ELSE
          CLSSUM3 = CLSSUM(CLASS)
         ENDIF
  322  CONTINUE
C
      ZP = (CLSSUM1 - CLSSUM2)**2
      Z2 = (LAMVAL(L)*TSS*SIGE) / (ZP*GAMVAL(K))
      Z3 = SQRT(Z2)
      DO 325 CLASS = 1,A
        IF (CLASS.EQ.1) THEN
        X2VEC(CLASS) = Z3
        ELSE IF (CLASS.EQ.2) THEN
        X2VEC(CLASS) = -Z3
        ELSE
        X2VEC(CLASS) = 0
        ENDIF
```

```
  325 CONTINUE
C
      SUM2 = 0
      DO 390 CLASS = 1,A
         DO 385 TIME = 1,B
            X2MAT(TIME,CLASS) = X2VEC(CLASS)
  385    CONTINUE
  390 CONTINUE
C
      DO 396 CLASS = 1,A
         X1CLSMN(CLASS) = CLSSUM(CLASS)/B
  396 CONTINUE
C
      BSS = 0
      DO 402 CLASS = 1,A
         DO 401 TIME = 1,B
            BSS = BSS + (X1CLSMN(CLASS)*X1CLSMN(CLASS))
  401    CONTINUE
  402 CONTINUE
C
      X2PX1 = 0
      DO 404 CLASS = 1,A
         DO 403 TIME = 1,B
            X2PX1 = X2PX1
     +          + (X2MAT(TIME,CLASS)*X1CMAT(TIME,CLASS))
  403    CONTINUE
  404 CONTINUE
C
      Z = (SIGE*TSS*TSS*LAMVAL(L))/((X2PX1*X2PX1)*BSS)
      BETA2 = SQRT(Z)
C
      LAMBDA = (BETA2*X2PX1*X2PX1*BETA2*BSS)/(SIGE*TSS*TSS)
C
      X2SSQ = 0
      X2STDSQ = 0
      VARX2STD = 0
      DO 410 CLASS = 1,A
         X2SSQ = X2SSQ + (B*(X2VEC(CLASS)**2))
  410 CONTINUE
C
      X2SSQQ = SQRT(X2SSQ)
      DO 415 CLASS = 1,A
         X2STD(CLASS) = X2VEC(CLASS)/X2SSQQ
         X2STDSQ = X2STDSQ + (B*(X2STD(CLASS)**2))
         VARX2STD = X2STDSQ / (N-1)
  415 CONTINUE
C
      IF (VARX2STD.GT.1.5) THEN
         COUNTER2 = COUNTER2 + 1
      ELSE
         COUNTER2 = COUNTER2
      ENDIF
C
     CALL RNNOA(A,ERVEC)
```

```
      CALL RNNOA(N,EVEC)
      COUNT3 = 1
      DO 435 CLASS = 1,A
         DO 434 TIME = 1,B
            EMAT(TIME,CLASS) = EVEC(COUNT3)
            COUNT3 = COUNT3 + 1
 434     CONTINUE
 435  CONTINUE
 440  CONTINUE
C
      TSSSQ = SQRT(TSS)
      DO 450 CLASS = 1,A
         X1MSTD(CLASS) = (CLSSUM(CLASS)/B)/TSSSQ
 450  CONTINUE
C
      DO 470 CLASS = 1,A
         DO 460 TIME = 1,B
            X1STD(TIME,CLASS) = X1CMAT(TIME,CLASS)/TSSSQ
 460     CONTINUE
 470  CONTINUE
C
      DO 510 CLASS = 1,A
         YB(CLASS) = BETA0 + (BETA1*X1MSTD(CLASS)) +
     +               (BETA2*X2STD(CLASS)) +
     +               (RHOSQ(W)*ERVEC(CLASS))
 510  CONTINUE
C
      DO 540 CLASS = 1,A
         DO 530 TIME = 1,B
            X1WISTD(TIME,CLASS) = X1STD(TIME,CLASS) -
                                  X1MSTD(CLASS)
 530     CONTINUE
 540  CONTINUE
C
      DO 580 CLASS = 1,A
         DO 570 TIME = 1,B
            YMAT(TIME,CLASS) = YB(CLASS) +
            (BETA1*X1WISTD(TIME,CLASS))   SIM03720
     +                           + EMAT(TIME,CLASS)
 570     CONTINUE
 580  CONTINUE
C
      X1PY = 0
      DO 620 CLASS = 1,A
         DO 610 TIME = 1,B
            X1PY = X1PY +
               (X1STD(TIME,CLASS)*YMAT(TIME,CLASS))
 610  CONTINUE
 620  CONTINUE
C
      B1HAT1 = X1PY
C
      X1PR1Y = 0
      DO 640 CLASS = 1,A
```

```
         DO 630 TIME = 1,B
            X1PR1Y = X1PR1Y +
                         (X1MSTD(CLASS)*YMAT(TIME,CLASS))
  630 CONTINUE
  640 CONTINUE
C
      B1BHAT = X1PR1Y/(BSS/TSS)
C
      X1PR2Y = 0
      DO 660 CLASS = 1,A
         DO 650 TIME = 1,B
            X1PR2Y = X1PR2Y +
                      (X1WISTD(TIME,CLASS)*YMAT(TIME,CLASS))
  650 CONTINUE
  660 CONTINUE
C
      B1WHAT = X1PR2Y/(WSS/TSS)
C
      SUMY = 0
      DO 680 CLASS = 1,A
         YCLSSUM(CLASS) = 0
         DO 670 TIME = 1,B
            SUMY = SUMY + YMAT(TIME,CLASS)
            YCLSSUM(CLASS) = YCLSSUM(CLASS) +
               YMAT(TIME,CLASS)
  670    CONTINUE
  680 CONTINUE
C
      DO 685 CLASS = 1,A
         YCLSMN(CLASS) = YCLSSUM(CLASS)/B
  685 CONTINUE
C
      YBAR = SUMY/(A*B)
      YPR2Y = 0
      DO 700 CLASS = 1,A
         DO 690 TIME = 1,B
            YWI(TIME,CLASS) = YMAT(TIME,CLASS) -
               YCLSMN(CLASS)
            YPR2Y = YPR2Y + YWI(TIME,CLASS)*YWI(TIME,CLASS))
  690    CONTINUE
  700 CONTINUE
C
      XWPXW = 0
      YPXW = 0
      DO 720 CLASS = 1,A
         DO 710 TIME = 1,B
            YPXW = YPXW +
                      (YMAT(TIME,CLASS)*X1WISTD(TIME,CLASS))
            XWPXW = XWPXW +
                   (X1WISTD(TIME,CLASS)*X1WISTD(TIME,CLASS))
            XWPIXW = 1/XWPXW
  710    CONTINUE
  720 CONTINUE
C
```

```
      YPXWY = (YPXW*YPXW*XWPIXW)
C
      SIGEHAT = (YPR2Y - YPXWY)/(A*(B-1)-1)
C
      YPBETY = 0
      DO 750 CLASS = 1,A
         YBET(CLASS) = YCLSMN(CLASS) - YBAR
         YPBETY = YPBETY + B*(YBET(CLASS)*YBET(CLASS))
  750 CONTINUE
C
      YPXB = 0
      XBPXB = 0
      DO 770 CLASS = 1,A
         DO 760 TIME = 1,B
            YPXB = YPXB + (YMAT(TIME,CLASS)*X1MSTD(CLASS))
            XBPXB = XBPXB + (X1MSTD(CLASS)*X1MSTD(CLASS))
            XBPIXB = 1/XBPXB
  760    CONTINUE
  770 CONTINUE
C
      YPXBY = YPXB*YPXB*XBPIXB
C
      SIGRHAT = (((YPBETY - YPXBY)/(A-2)) - SIGEHAT)/B
      COUNTER3 = 0
      IF (SIGRHAT.LT.0) THEN
         RHOHAT = 0
         COUNTER3 = COUNTER3 + 1
      ELSE
         RHOHAT = SIGRHAT/SIGEHAT
      ENDIF
C
      D1 = BSS/TSS
      D2 = WSS/TSS
      D3 = 1/((B*SIGRHAT) + SIGEHAT)
      D4 = 1/SIGEHAT
      B1HAT2 = ((D1*B1BHAT) + (D2*B1WHAT))/(D1 + D2)
      B1STRHAT = ((D1*D3*B1BHAT) + (D2*D4*B1WHAT))/((D1*D3)
                 + (D2*D4))
C
      RELEFF = (((1+(B*RHOHAT*GAMVAL(K)))*
                (1+(B*RHOHAT))*GAMVAL(K))
     +           + LAMVAL(L))/((1+(B*RHOHAT*GAMVAL(K)))**2)
C
      WRITE(4,*) LAMVAL(L),RHOVAL(W),RELEFF
  830 CONTINUE
C
  840 CONTINUE
C
  850 CONTINUE
C
  860 CONTINUE
      WRITE(4,*) 'CVARX2',COUNTER2,'CRNEG',COUNTER3
C
  900 END
```

Program 2 for a = 5

The following is an example of Program 2 for a = 5, b = 2, and γ = .2. The a, b, and γ value may be changed for the other parameter combinations utilizing Program 2 for a = 5.

```
      INTEGER A,B,N,CLASS,TIME,COUNT,SIZE,PASS,FC,COUNTER2,
         COUNT3,W
      INTEGER NUM,NUM2,COUNTER3
      REAL XMAT(2,5),XVEC(10),SUM,SUMCOL,XDOTI(5),XBAR,TSS2
      REAL WSS,TSS,GAMMA,SMAT(2,5),ZMAT(2,5),GAMVAL(5),
         LAMBDA
      REAL DELTA,DELTAN,DELTA0,CHANGE(5),NXBAR,EPS,X2SSQQ,
         X2PX1
      REAL X2VEC(5),X2MAT(2,5),X1CMAT(2,5),X2SUM,X1CLSMN(3)
      REAL SUM2,CLSSUM(5),Z,X2STD(5),X2SSQ,X2STDSQ,X2STDSUM
      REAL LAMVAL(5),ERVEC(5),EVEC(10),EMAT(2,5),SIGE,SIGR,
         BSS
      REAL X1STD(2,5),X1MSTD(5),BETA0,BETA1,BETA2,
         X1WISTD(2,5),BSS2
      REAL YB(5),YMAT(2,5),RHOVAL(5),RHOHAT,TSSSQ,
         RHOSQ(5),XWPXW
      REAL B1WHAT,B1BHAT,B1HAT1,B1STRHAT,SIGEHAT,
         SIGRHAT,YVEC(10)
      REAL RELEFF,X1PY,X1PR1Y,SUMY,YCLSSUM(5),YBAR,YPR2Y,
         B1HAT2
      REAL YCLSMN(5),XWPIXW,YPXW,YPXWY,YPBETY,YPXB,XBPXB,
         XBPIXB
      REAL YBET(5),YWI(2,5)
      A=5
      B=2
      N=A*B
C
      EPS = 0.00005
      GAMVAL(1)  = .001
      GAMVAL(2)  = .2
      GAMVAL(3)  = .5
      GAMVAL(4)  = .7
      GAMVAL(5)  = .99
      LAMVAL(1)  = 0
      LAMVAL(2)  = .5
      LAMVAL(3)  = 1
      LAMVAL(4)  = 2
      LAMVAL(5)  = 5
      RHOVAL(1)  = 0
      RHOVAL(2)  = .5
      RHOVAL(3)  = 1
      RHOVAL(4)  = 2
      RHOVAL(5)  = 5
      RHOSQ(1)  = SQRT(RHOVAL(1))
      RHOSQ(2)  = SQRT(RHOVAL(2))
```

```
      RHOSQ(3) = SQRT(RHOVAL(3))
      RHOSQ(4) = SQRT(RHOVAL(4))
      RHOSQ(5) = SQRT(RHOVAL(5))
      X2VEC(1) = 0
      X2VEC(2) = 0
      X2VEC(3) = 0
      X2VEC(4) = 0
      X2VEC(5) = 0
      SIGE = 1
      BETA0 = 0
      BETA1 = 1
C
      WRITE(4,*) 'A',A,'B',B
      COUNT2 = 0
      COUNTER3 = 0
C
      CALL RNOPT(2)
      CALL RNSET(262511)
C
      DO 860 M = 1,1000
C
      DO 850 K = 2,2
C
      DO 840 L = 1,5
C
      DO 830 W = 1,5
C
      LOOPD = 0
    5 LOOPC = 0
      LOOPD = LOOPD + 1
      IF (LOOPD.LE.50) THEN
        GO TO 6
      ELSE
        WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
        GO TO 860
      ENDIF

      DO 250 J = 1,50
C
    6 RNUN(N,XVEC)
      DO 7 I = 1,N
         XVEC(I) = XVEC(I) * 15
    7 CONTINUE
C
      SUM = 0
      COUNT = 1
      DO 20 CLASS = 1,A
            SUMCOL = 0
            DO 10 TIME = 1,B
                XMAT(TIME,CLASS) = XVEC(COUNT)
                SUM = SUM + XMAT(TIME,CLASS)
                SUMCOL = SUMCOL + XMAT(TIME,CLASS)
                COUNT = COUNT + 1
   10       CONTINUE
```

```
            XDOTI(CLASS) = SUMCOL/B
   20 CONTINUE
C
      XBAR = SUM/N
      WSS = 0
      TSS = 0
C
      DO 40 CLASS = 1,A
         DO 30 TIME = 1,B
            SMAT(TIME,CLASS) = XMAT(TIME,CLASS) -
            XDOTI(CLASS)
            WSS = WSS +
            (SMAT(TIME,CLASS))*(SMAT(TIME,CLASS))
            TSS = TSS + (XMAT(TIME,CLASS) - XBAR)*
     +                              (XMAT(TIME,CLASS) - XBAR)
   30    CONTINUE
   40 CONTINUE
C
   50 GAMMA = WSS/TSS
      SUMNEW = 0
C
      IF ((GAMMA.GE.GAMVAL(K)-
         EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +   THEN
         GO TO 300
      ENDIF
C
      IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
         SIZE = 0
         FC = 0
         PASS = 1
      ENDIF
C
      IF (GAMMA.GT.GAMVAL(K)+EPS) THEN
         SIZE = 1
         FC = 0
         PASS = 1
      ENDIF
C
      DELTA = .5
      NXBAR = XBAR
C
   55 LOOPC = LOOPC + 1
      IF (LOOPC.LE.50) THEN
        GO TO 56
      ELSE
        GO TO 5
      ENDIF
   56 WSS = 0
      TSS = 0
      IF ((SIZE.EQ.0).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = .75
         ELSE
```

```
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 70 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 60 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
 60         CONTINUE
 70      CONTINUE
         DO 90 CLASS = 1,A
            DO 80 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
   +                        (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
   +                        (ZMAT(TIME,CLASS) - NXBAR)
 80             CONTINUE
 90      CONTINUE
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
   +        THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 0
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
         GO TO 55
C
      ELSE IF ((SIZE.EQ.1).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = 1.5
         ELSE
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 + DELTA
         NXBAR = DELTAN*XBAR
         DO 110 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 100 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
100         CONTINUE
110      CONTINUE
         DO 130 CLASS = 1,A
            DO 120 TIME = 1,B
```

```
              WSS = WSS + (ZMAT(TIME,CLASS) -
                           CHANGE(CLASS))*
     +                     (ZMAT(TIME,CLASS) -
                           CHANGE(CLASS))
              TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                     (ZMAT(TIME,CLASS) - NXBAR)
 120        CONTINUE
 130      CONTINUE
          GAMMA = WSS/TSS
          IF ((GAMMA.GE.GAMVAL(K)-
             EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +       THEN
             GO TO 300
          ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
             SIZE = 0
             FC = 1
          ELSE
             SIZE = 1
             FC = 0
          ENDIF
          GO TO 55
C
       ELSE IF ((SIZE.EQ.0).AND.(FC.EQ.1)) THEN
          DELTA0 = DELTAN
          DELTA = DELTA*(0.7)
          DELTAN = DELTA0 - DELTA
          NXBAR = DELTAN*XBAR
          DO 150 CLASS = 1,A
             CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
             DO 140 TIME = 1,B
                ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                   CHANGE(CLASS)
 140         CONTINUE
 150      CONTINUE
          DO 170 CLASS = 1,A
             DO 160 TIME = 1,B
                WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                       (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
                TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                       (ZMAT(TIME,CLASS) - NXBAR)
 160         CONTINUE
 170      CONTINUE
          GAMMA = WSS/TSS
          IF ((GAMMA.GE.GAMVAL(K)-
             EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +       THEN
             GO TO 300
          ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
             SIZE = 0
             FC = 1
          ELSE
             SIZE = 1
```

```
            FC = 1
         ENDIF
         GO TO 55
C
      ELSE
         DELTA0 = DELTAN
         DELTA = DELTA*(0.7)
         DELTAN = DELTA0 + DELTA
         NXBAR = DELTAN*XBAR
         DO 190 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 180 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
  180       CONTINUE
  190    CONTINUE
         DO 210 CLASS = 1,A
            DO 200 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))*
     +                     (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                     (ZMAT(TIME,CLASS) - NXBAR)
  200       CONTINUE
  210    CONTINUE
C
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 1
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
         GO TO 55
C
      ENDIF
C
  250 CONTINUE
C
  300 IF ((GAMMA.GE.GAMVAL(K)-
           EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +     THEN
           GO TO 305
      ELSE
           WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
      ENDIF
C
  305 DO 320 CLASS = 1,A
```

```
         CLSSUM(CLASS) = 0
         DO 310 TIME = 1,B
            X1CMAT(TIME,CLASS) = ZMAT(TIME,CLASS) - NXBAR
            CLSSUM(CLASS) = CLSSUM(CLASS) +
                            X1CMAT(TIME,CLASS)
  310    CONTINUE
  320 CONTINUE
C
      CLSSUM1 = 0
      CLSSUM2 = 0
      CLSSUM3 = 0
      CLSSUM4 = 0
      CLSSUM5 = 0
      DO 322 CLASS = 1,A
         IF (CLASS.EQ.1) THEN
          CLSSUM1 = CLSSUM(CLASS)
         ELSE IF (CLASS.EQ.2) THEN
          CLSSUM2 = CLSSUM(CLASS)
          ELSE IF (CLASS.EQ.3) THEN
          CLSSUM3 = CLSSUM(CLASS)
          ELSE IF (CLASS.EQ.4) THEN
          CLSSUM4 = CLSSUM(CLASS)
         ELSE
          CLSSUM5 = CLSSUM(CLASS)
         ENDIF
  322  CONTINUE
C
      ZP = (CLSSUM1 - CLSSUM2)**2
      Z2 = (LAMVAL(L)*TSS*SIGE) / (ZP*GAMVAL(K))
      Z3 = SQRT(Z2)
      DO 325 CLASS = 1,A
        IF (CLASS.EQ.1) THEN
        X2VEC(CLASS) = Z3
        ELSE IF (CLASS.EQ.2) THEN
        X2VEC(CLASS) = -Z3
        ELSE IF (CLASS.EQ.3) THEN
        X2VEC(CLASS) = Z3/2
        ELSE IF (CLASS.EQ.4) THEN
        X2VEC(CLASS) = -Z3/2
        ELSE
        X2VEC(CLASS) = 0
        ENDIF
  325 CONTINUE
C
       SUM2 = 0
       DO 390 CLASS = 1,A
          DO 385 TIME = 1,B
             X2MAT(TIME,CLASS) = X2VEC(CLASS)
  385     CONTINUE
  390  CONTINUE
C
       DO 396 CLASS = 1,A
          X1CLSMN(CLASS) = CLSSUM(CLASS)/B
  396  CONTINUE
```

```
C
      BSS = 0
      DO 402 CLASS = 1,A
         DO 401 TIME = 1,B
            BSS = BSS + (X1CLSMN(CLASS)*X1CLSMN(CLASS))
 401     CONTINUE
 402  CONTINUE
C
      X2PX1 = 0
      DO 404 CLASS = 1,A
         DO 403 TIME = 1,B
            X2PX1 = X2PX1
     +           + (X2MAT(TIME,CLASS)*X1CMAT(TIME,CLASS))
 403     CONTINUE
 404  CONTINUE
C
      Z = (SIGE*TSS*TSS*LAMVAL(L))/((X2PX1*X2PX1)*BSS)
      BETA2 = SQRT(Z)
C
      LAMBDA = (BETA2*X2PX1*X2PX1*BETA2*BSS)/(SIGE*TSS*TSS)
C
      X2SSQ = 0
      X2STDSQ = 0
      VARX2STD = 0
      DO 410 CLASS = 1,A
         X2SSQ = X2SSQ + (B*(X2VEC(CLASS)**2))
 410  CONTINUE
C
      X2SSQQ = SQRT(X2SSQ)
      DO 415 CLASS = 1,A
         X2STD(CLASS) = X2VEC(CLASS)/X2SSQQ
         X2STDSQ = X2STDSQ + (B*(X2STD(CLASS)**2))
         VARX2STD = X2STDSQ / (N-1)
 415  CONTINUE
C
      IF (VARX2STD.GT.1.5) THEN
         COUNTER2 = COUNTER2 + 1
      ELSE
         COUNTER2 = COUNTER2
      ENDIF
C
     CALL RNNOA(A,ERVEC)
     CALL RNNOA(N,EVEC)
     COUNT3 = 1
     DO 435 CLASS = 1,A
        DO 434 TIME = 1,B
           EMAT(TIME,CLASS) = EVEC(COUNT3)
           COUNT3 = COUNT3 + 1
 434    CONTINUE
 435 CONTINUE
 440 CONTINUE
C
     TSSSQ = SQRT(TSS)
     DO 450 CLASS = 1,A
```

```
         X1MSTD(CLASS) = (CLSSUM(CLASS)/B)/TSSSQ
  450 CONTINUE
C
      DO 470 CLASS = 1,A
         DO 460 TIME = 1,B
            X1STD(TIME,CLASS) = X1CMAT(TIME,CLASS)/TSSSQ
  460    CONTINUE
  470 CONTINUE
C
      DO 510 CLASS = 1,A
         YB(CLASS) = BETA0 + (BETA1*X1MSTD(CLASS)) +
     +               (BETA2*X2STD(CLASS)) +
     +               (RHOSQ(W)*ERVEC(CLASS))
  510 CONTINUE
C
      DO 540 CLASS = 1,A
         DO 530 TIME = 1,B
            X1WISTD(TIME,CLASS) = X1STD(TIME,CLASS) -
                                  X1MSTD(CLASS)
  530    CONTINUE
  540 CONTINUE
C
      DO 580 CLASS = 1,A
         DO 570 TIME = 1,B
            YMAT(TIME,CLASS) = YB(CLASS) +
            (BETA1*X1WISTD(TIME,CLASS))  SIM03720
     +                               + EMAT(TIME,CLASS)
  570    CONTINUE
  580 CONTINUE
C
      X1PY = 0
      DO 620 CLASS = 1,A
         DO 610 TIME = 1,B
            X1PY = X1PY +
               (X1STD(TIME,CLASS)*YMAT(TIME,CLASS))
  610 CONTINUE
  620 CONTINUE
C
      B1HAT1 = X1PY
C
      X1PR1Y = 0
      DO 640 CLASS = 1,A
         DO 630 TIME = 1,B
            X1PR1Y = X1PR1Y +
                        (X1MSTD(CLASS)*YMAT(TIME,CLASS))
  630 CONTINUE
  640 CONTINUE
C
      B1BHAT = X1PR1Y/(BSS/TSS)
C
      X1PR2Y = 0
      DO 660 CLASS = 1,A
         DO 650 TIME = 1,B
            X1PR2Y = X1PR2Y +
```

```
                         (X1WISTD(TIME,CLASS)*YMAT(TIME,CLASS))
  650 CONTINUE
  660 CONTINUE
C
      B1WHAT = X1PR2Y/(WSS/TSS)
C
      SUMY = 0
      DO 680 CLASS = 1,A
         YCLSSUM(CLASS) = 0
         DO 670 TIME = 1,B
            SUMY = SUMY + YMAT(TIME,CLASS)
            YCLSSUM(CLASS) = YCLSSUM(CLASS) +
               YMAT(TIME,CLASS)
  670    CONTINUE
  680 CONTINUE
C
      DO 685 CLASS = 1,A
         YCLSMN(CLASS) = YCLSSUM(CLASS)/B
  685 CONTINUE
C
      YBAR = SUMY/(A*B)
      YPR2Y = 0
      DO 700 CLASS = 1,A
         DO 690 TIME = 1,B
            YWI(TIME,CLASS) = YMAT(TIME,CLASS) -
               YCLSMN(CLASS)
            YPR2Y = YPR2Y + YWI(TIME,CLASS)*YWI(TIME,CLASS))
  690    CONTINUE
  700 CONTINUE
C
      XWPXW = 0
      YPXW = 0
      DO 720 CLASS = 1,A
         DO 710 TIME = 1,B
            YPXW = YPXW +
                         (YMAT(TIME,CLASS)*X1WISTD(TIME,CLASS))
            XWPXW = XWPXW +
                      (X1WISTD(TIME,CLASS)*X1WISTD(TIME,CLASS))
            XWPIXW = 1/XWPXW
  710    CONTINUE
  720 CONTINUE
C
      YPXWY = (YPXW*YPXW*XWPIXW)
C
      SIGEHAT = (YPR2Y - YPXWY)/(A*(B-1)-1)
C
      YPBETY = 0
      DO 750 CLASS = 1,A
         YBET(CLASS) = YCLSMN(CLASS) - YBAR
         YPBETY = YPBETY + B*(YBET(CLASS)*YBET(CLASS))
  750 CONTINUE
C
      YPXB = 0
      XBPXB = 0
```

```
      DO 770 CLASS = 1,A
         DO 760 TIME = 1,B
            YPXB = YPXB + (YMAT(TIME,CLASS)*X1MSTD(CLASS))
            XBPXB = XBPXB + (X1MSTD(CLASS)*X1MSTD(CLASS))
            XBPIXB = 1/XBPXB
  760    CONTINUE
  770 CONTINUE
C
      YPXBY = YPXB*YPXB*XBPIXB
C
      SIGRHAT = (((YPBETY - YPXBY)/(A-2)) - SIGEHAT)/B
      COUNTER3 = 0
      IF (SIGRHAT.LT.0) THEN
         RHOHAT = 0
         COUNTER3 = COUNTER3 + 1
      ELSE
         RHOHAT = SIGRHAT/SIGEHAT
      ENDIF
C
      D1 = BSS/TSS
      D2 = WSS/TSS
      D3 = 1/((B*SIGRHAT) + SIGEHAT)
      D4 = 1/SIGEHAT
      B1HAT2 = ((D1*B1BHAT) + (D2*B1WHAT))/(D1 + D2)
      B1STRHAT = ((D1*D3*B1BHAT) + (D2*D4*B1WHAT))/((D1*D3)
                 + (D2*D4))
C
      RELEFF = (((1+(B*RHOHAT*GAMVAL(K)))*
                (1+(B*RHOHAT))*GAMVAL(K))
     +           + LAMVAL(L))/((1+(B*RHOHAT*GAMVAL(K)))**2)
C
      WRITE(4,*) LAMVAL(L),RHOVAL(W),RELEFF
  830 CONTINUE
C
  840 CONTINUE
C
  850 CONTINUE
C
  860 CONTINUE
      WRITE(4,*) 'CVARX2',COUNTER2,'CRNEG',COUNTER3
C
  900 END
```

Program 3 for a = 3

The following is an example of Program 3 for a = 3, b = 2, and γ = .99. The a, b, and γ value may be changed for the other parameter combinations utilizing Program 3 for a = 3.

```
      INTEGER A,B,N,CLASS,TIME,COUNT,SIZE,PASS,FC,COUNTER2,
         COUNT3,W
      INTEGER NUM,NUM2,COUNTER3
```

```
      REAL XMAT(2,3),XVEC(6),SUM,SUMCOL,XDOTI(3),XBAR,TSS2
      REAL WSS,TSS,GAMMA,SMAT(2,3),ZMAT(2,3),GAMVAL(5),
         LAMBDA
      REAL DELTA,DELTAN,DELTA0,CHANGE(3),NXBAR,EPS,X2SSQQ,
         X2PX1
      REAL X2VEC(3),X2MAT(2,3),X1CMAT(2,3),X2SUM,X1CLSMN(3)
      REAL SUM2,CLSSUM(3),Z,X2STD(3),X2SSQ,X2STDSQ,X2STDSUM
      REAL LAMVAL(5),ERVEC(3),EVEC(6),EMAT(2,3),SIGE,SIGR,
         BSS
      REAL X1STD(2,3),X1MSTD(3),BETA0,BETA1,BETA2,
         X1WISTD(2,3),BSS2
      REAL YB(3),YMAT(2,3),RHOVAL(5),RHOHAT,TSSSQ,
         RHOSQ(5),XWPXW
      REAL B1WHAT,B1BHAT,B1HAT1,B1STRHAT,SIGEHAT,
         SIGRHAT,YVEC(6)
      REAL RELEFF,X1PY,X1PR1Y,SUMY,YCLSSUM(3),YBAR,YPR2Y,
         B1HAT2
      REAL YCLSMN(3),XWPIXW,YPXW,YPXWY,YPBETY,YPXB,XBPXB,
         XBPIXB
      REAL YBET(3),YWI(2,3)
      A=3
      B=2
      N=A*B
C
      EPS = 0.00005
      GAMVAL(1) = .001
      GAMVAL(2) = .2
      GAMVAL(3) = .5
      GAMVAL(4) = .7
      GAMVAL(5) = .99
      LAMVAL(1) = 0
      LAMVAL(2) = .5
      LAMVAL(3) = 1
      LAMVAL(4) = 2
      LAMVAL(5) = 5
      RHOVAL(1) = 0
      RHOVAL(2) = .5
      RHOVAL(3) = 1
      RHOVAL(4) = 2
      RHOVAL(5) = 5
      RHOSQ(1) = SQRT(RHOVAL(1))
      RHOSQ(2) = SQRT(RHOVAL(2))
      RHOSQ(3) = SQRT(RHOVAL(3))
      RHOSQ(4) = SQRT(RHOVAL(4))
      RHOSQ(5) = SQRT(RHOVAL(5))
      X2VEC(1) = 0
      X2VEC(2) = 0
      X2VEC(3) = 0
      SIGE = 1
      BETA0 = 0
      BETA1 = 1
C
      WRITE(4,*) 'A',A,'B',B
      COUNT2 = 0
```

```
      COUNTER3 = 0
C
      CALL RNOPT(2)
      CALL RNSET(351)
C
      DO 860 M = 1,250
C
      DO 850 K = 5,5
C
      DO 840 L = 1,5
C
      DO 830 W = 1,5
C
      LOOPD = 0
    5 LOOPC = 0
      LOOPD = LOOPD + 1
      IF (LOOPD.LE.50) THEN
        GO TO 6
      ELSE
        WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
        GO TO 860
      ENDIF

      DO 250 J = 1,50
C
    6 RNUN(N,XVEC)
      DO 7 I = 1,N
         XVEC(I) = XVEC(I) * 15
    7 CONTINUE
C
      SUM = 0
      COUNT = 1
      DO 20 CLASS = 1,A
            SUMCOL = 0
            DO 10 TIME = 1,B
                XMAT(TIME,CLASS) = XVEC(COUNT)
                SUM = SUM + XMAT(TIME,CLASS)
                SUMCOL = SUMCOL + XMAT(TIME,CLASS)
                COUNT = COUNT + 1
   10       CONTINUE
            XDOTI(CLASS) = SUMCOL/B
   20 CONTINUE
C
      XBAR = SUM/N
      WSS = 0
      TSS = 0
C
      DO 40 CLASS = 1,A
         DO 30 TIME = 1,B
            SMAT(TIME,CLASS) = XMAT(TIME,CLASS) -
            XDOTI(CLASS)
            WSS = WSS +
            (SMAT(TIME,CLASS))*(SMAT(TIME,CLASS))
            TSS = TSS + (XMAT(TIME,CLASS) - XBAR)*
```

```
     +                              (XMAT(TIME,CLASS) - XBAR)
   30      CONTINUE
   40 CONTINUE
C
   50 GAMMA = WSS/TSS
      SUMNEW = 0
C
      IF ((GAMMA.GE.GAMVAL(K)-
         EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +   THEN
         GO TO 300
      ENDIF
C
      IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
         SIZE = 0
         FC = 0
         PASS = 1
      ENDIF
C
      IF (GAMMA.GT.GAMVAL(K)+EPS) THEN
         SIZE = 1
         FC = 0
         PASS = 1
      ENDIF
C
      DELTA = .5
      NXBAR = XBAR
C
   55 LOOPC = LOOPC + 1
      IF (LOOPC.LE.50) THEN
        GO TO 56
      ELSE
        GO TO 5
      ENDIF
   56 WSS = 0
      TSS = 0
      IF ((SIZE.EQ.0).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = .75
         ELSE
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 70 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 60 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
   60       CONTINUE
   70    CONTINUE
         DO 90 CLASS = 1,A
            DO 80 TIME = 1,B
```

```
                  WSS = WSS + (ZMAT(TIME,CLASS) -
                               CHANGE(CLASS))*
     +                        (ZMAT(TIME,CLASS) -
                               CHANGE(CLASS))
                  TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                        (ZMAT(TIME,CLASS) - NXBAR)
 80            CONTINUE
 90         CONTINUE
            GAMMA = WSS/TSS
            IF ((GAMMA.GE.GAMVAL(K)-
               EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +         THEN
               GO TO 300
            ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
               SIZE = 0
               FC = 0
            ELSE
               SIZE = 1
               FC = 1
            ENDIF
            GO TO 55
C
         ELSE IF ((SIZE.EQ.1).AND.(FC.EQ.0)) THEN
            IF (PASS.EQ.1) THEN
               DELTA0 = 1.5
            ELSE
               DELTA0 = DELTAN
            ENDIF
            PASS = 0
            DELTAN = DELTA0 + DELTA
            NXBAR = DELTAN*XBAR
            DO 110 CLASS = 1,A
               CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
               DO 100 TIME = 1,B
                  ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                     CHANGE(CLASS)
 100           CONTINUE
 110        CONTINUE
            DO 130 CLASS = 1,A
               DO 120 TIME = 1,B
                  WSS = WSS + (ZMAT(TIME,CLASS) -
                               CHANGE(CLASS))*
     +                        (ZMAT(TIME,CLASS) -
                               CHANGE(CLASS))
                  TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                        (ZMAT(TIME,CLASS) - NXBAR)
 120           CONTINUE
 130        CONTINUE
            GAMMA = WSS/TSS
            IF ((GAMMA.GE.GAMVAL(K)-
               EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +         THEN
               GO TO 300
            ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
```

```
            SIZE = 0
            FC = 1
         ELSE
            SIZE = 1
            FC = 0
         ENDIF
         GO TO 55
C
      ELSE IF ((SIZE.EQ.0).AND.(FC.EQ.1)) THEN
         DELTA0 = DELTAN
         DELTA = DELTA*(0.5)
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 150 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 140 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
 140        CONTINUE
 150     CONTINUE
         DO 170 CLASS = 1,A
            DO 160 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))*
     +                     (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                     (ZMAT(TIME,CLASS) - NXBAR)
 160        CONTINUE
 170     CONTINUE
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 1
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
         GO TO 55
C
      ELSE
         DELTA0 = DELTAN
         DELTA = DELTA*(0.5)
         DELTAN = DELTA0 + DELTA
         NXBAR = DELTAN*XBAR
         DO 190 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 180 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
```

```
  180        CONTINUE
  190     CONTINUE
          DO 210 CLASS = 1,A
             DO 200 TIME = 1,B
                WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
                TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
  200        CONTINUE
  210     CONTINUE
C
          GAMMA = WSS/TSS
          IF ((GAMMA.GE.GAMVAL(K)-
             EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +       THEN
             GO TO 300
          ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
             SIZE = 0
             FC = 1
          ELSE
             SIZE = 1
             FC = 1
          ENDIF
          GO TO 55
C
       ENDIF
C
  250 CONTINUE
C
  300 IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 305
      ELSE
            WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
      ENDIF
C
  305 DO 320 CLASS = 1,A
          CLSSUM(CLASS) = 0
          DO 310 TIME = 1,B
             X1CMAT(TIME,CLASS) = ZMAT(TIME,CLASS) - NXBAR
             CLSSUM(CLASS) = CLSSUM(CLASS) +
                             X1CMAT(TIME,CLASS)
  310     CONTINUE
  320 CONTINUE
C
      CLSSUM1 = 0
      CLSSUM2 = 0
      CLSSUM3 = 0
      CLSSUM4 = 0
      CLSSUM5 = 0
      DO 322 CLASS = 1,A
```

```
        IF (CLASS.EQ.1) THEN
         CLSSUM1 = CLSSUM(CLASS)
        ELSE IF (CLASS.EQ.2) THEN
         CLSSUM2 = CLSSUM(CLASS)
        ELSE
         CLSSUM3 = CLSSUM(CLASS)
        ENDIF
 322  CONTINUE
C
      ZP = (CLSSUM1 - CLSSUM2)**2
      Z2 = (LAMVAL(L)*TSS*SIGE) / (ZP*GAMVAL(K))
      Z3 = SQRT(Z2)
      DO 325 CLASS = 1,A
        IF (CLASS.EQ.1) THEN
        X2VEC(CLASS) = Z3
        ELSE IF (CLASS.EQ.2) THEN
        X2VEC(CLASS) = -Z3
        ELSE
        X2VEC(CLASS) = 0
        ENDIF
 325 CONTINUE
C
      SUM2 = 0
      DO 390 CLASS = 1,A
         DO 385 TIME = 1,B
            X2MAT(TIME,CLASS) = X2VEC(CLASS)
 385     CONTINUE
 390  CONTINUE
C
      DO 396 CLASS = 1,A
         X1CLSMN(CLASS) = CLSSUM(CLASS)/B
 396  CONTINUE
C
      BSS = 0
      DO 402 CLASS = 1,A
         DO 401 TIME = 1,B
            BSS = BSS + (X1CLSMN(CLASS)*X1CLSMN(CLASS))
 401     CONTINUE
 402  CONTINUE
C
      X2PX1 = 0
      DO 404 CLASS = 1,A
         DO 403 TIME = 1,B
            X2PX1 = X2PX1
     +          + (X2MAT(TIME,CLASS)*X1CMAT(TIME,CLASS))
 403     CONTINUE
 404  CONTINUE
C
      Z = (SIGE*TSS*TSS*LAMVAL(L))/((X2PX1*X2PX1)*BSS)
      BETA2 = SQRT(Z)
C
      LAMBDA = (BETA2*X2PX1*X2PX1*BETA2*BSS)/(SIGE*TSS*TSS)
C
      X2SSQ = 0
```

```
      X2STDSQ = 0
      VARX2STD = 0
      DO 410 CLASS = 1,A
         X2SSQ = X2SSQ + (B*(X2VEC(CLASS)**2))
 410  CONTINUE
C
      X2SSQQ = SQRT(X2SSQ)
      DO 415 CLASS = 1,A
         X2STD(CLASS) = X2VEC(CLASS)/X2SSQQ
         X2STDSQ = X2STDSQ + (B*(X2STD(CLASS)**2))
         VARX2STD = X2STDSQ / (N-1)
 415  CONTINUE
C
      IF (VARX2STD.GT.1.5) THEN
         COUNTER2 = COUNTER2 + 1
      ELSE
         COUNTER2 = COUNTER2
      ENDIF
C
     CALL RNNOA(A,ERVEC)
     CALL RNNOA(N,EVEC)
     COUNT3 = 1
     DO 435 CLASS = 1,A
        DO 434 TIME = 1,B
           EMAT(TIME,CLASS) = EVEC(COUNT3)
           COUNT3 = COUNT3 + 1
 434    CONTINUE
 435 CONTINUE
 440 CONTINUE
C
     TSSSQ = SQRT(TSS)
     DO 450 CLASS = 1,A
        X1MSTD(CLASS) = (CLSSUM(CLASS)/B)/TSSSQ
 450 CONTINUE
C
     DO 470 CLASS = 1,A
        DO 460 TIME = 1,B
           X1STD(TIME,CLASS) = X1CMAT(TIME,CLASS)/TSSSQ
 460    CONTINUE
 470 CONTINUE
C
     DO 510 CLASS = 1,A
        YB(CLASS) = BETA0 + (BETA1*X1MSTD(CLASS)) +
    +               (BETA2*X2STD(CLASS)) +
    +               (RHOSQ(W)*ERVEC(CLASS))
 510 CONTINUE
C
     DO 540 CLASS = 1,A
        DO 530 TIME = 1,B
           X1WISTD(TIME,CLASS) = X1STD(TIME,CLASS) -
                                 X1MSTD(CLASS)
 530    CONTINUE
 540 CONTINUE
C
```

```
      DO 580 CLASS = 1,A
         DO 570 TIME = 1,B
            YMAT(TIME,CLASS) = YB(CLASS) +
            (BETA1*X1WISTD(TIME,CLASS))   SIM03720
     +                              + EMAT(TIME,CLASS)
  570    CONTINUE
  580 CONTINUE
C
      X1PY = 0
      DO 620 CLASS = 1,A
         DO 610 TIME = 1,B
            X1PY = X1PY +
               (X1STD(TIME,CLASS)*YMAT(TIME,CLASS))
  610 CONTINUE
  620 CONTINUE
C
      B1HAT1 = X1PY
C
      X1PR1Y = 0
      DO 640 CLASS = 1,A
         DO 630 TIME = 1,B
            X1PR1Y = X1PR1Y +
                         (X1MSTD(CLASS)*YMAT(TIME,CLASS))
  630 CONTINUE
  640 CONTINUE
C
      B1BHAT = X1PR1Y/(BSS/TSS)
C
      X1PR2Y = 0
      DO 660 CLASS = 1,A
         DO 650 TIME = 1,B
            X1PR2Y = X1PR2Y +
                      (X1WISTD(TIME,CLASS)*YMAT(TIME,CLASS))
  650 CONTINUE
  660 CONTINUE
C
      B1WHAT = X1PR2Y/(WSS/TSS)
C
      SUMY = 0
      DO 680 CLASS = 1,A
         YCLSSUM(CLASS) = 0
         DO 670 TIME = 1,B
            SUMY = SUMY + YMAT(TIME,CLASS)
            YCLSSUM(CLASS) = YCLSSUM(CLASS) +
               YMAT(TIME,CLASS)
  670    CONTINUE
  680 CONTINUE
C
      DO 685 CLASS = 1,A
         YCLSMN(CLASS) = YCLSSUM(CLASS)/B
  685 CONTINUE
C
      YBAR = SUMY/(A*B)
      YPR2Y = 0
```

```
      DO 700 CLASS = 1,A
         DO 690 TIME = 1,B
            YWI(TIME,CLASS) = YMAT(TIME,CLASS) -
               YCLSMN(CLASS)
            YPR2Y = YPR2Y + YWI(TIME,CLASS)*YWI(TIME,CLASS))
  690    CONTINUE
  700 CONTINUE
C
      XWPXW = 0
      YPXW = 0
      DO 720 CLASS = 1,A
         DO 710 TIME = 1,B
            YPXW = YPXW +
                       (YMAT(TIME,CLASS)*X1WISTD(TIME,CLASS))
            XWPXW = XWPXW +
                    (X1WISTD(TIME,CLASS)*X1WISTD(TIME,CLASS))
            XWPIXW = 1/XWPXW
  710    CONTINUE
  720 CONTINUE
C
      YPXWY = (YPXW*YPXW*XWPIXW)
C
      SIGEHAT = (YPR2Y - YPXWY)/(A*(B-1)-1)
C
      YPBETY = 0
      DO 750 CLASS = 1,A
         YBET(CLASS) = YCLSMN(CLASS) - YBAR
         YPBETY = YPBETY + B*(YBET(CLASS)*YBET(CLASS))
  750 CONTINUE
C
      YPXB = 0
      XBPXB = 0
      DO 770 CLASS = 1,A
         DO 760 TIME = 1,B
            YPXB = YPXB + (YMAT(TIME,CLASS)*X1MSTD(CLASS))
            XBPXB = XBPXB + (X1MSTD(CLASS)*X1MSTD(CLASS))
            XBPIXB = 1/XBPXB
  760    CONTINUE
  770 CONTINUE
C
      YPXBY = YPXB*YPXB*XBPIXB
C
      SIGRHAT = (((YPBETY - YPXBY)/(A-2)) - SIGEHAT)/B
      COUNTER3 = 0
      IF (SIGRHAT.LT.0) THEN
         RHOHAT = 0
         COUNTER3 = COUNTER3 + 1
      ELSE
         RHOHAT = SIGRHAT/SIGEHAT
      ENDIF
C
      D1 = BSS/TSS
      D2 = WSS/TSS
      D3 = 1/((B*SIGRHAT) + SIGEHAT)
```

```
      D4 = 1/SIGEHAT
      B1HAT2 = ((D1*B1BHAT) + (D2*B1WHAT))/(D1 + D2)
      B1STRHAT = ((D1*D3*B1BHAT) + (D2*D4*B1WHAT))/((D1*D3)
                  + (D2*D4))
C
      RELEFF = (((1+(B*RHOHAT*GAMVAL(K)))*
                 (1+(B*RHOHAT))*GAMVAL(K))
     +            + LAMVAL(L))/((1+(B*RHOHAT*GAMVAL(K)))**2)
C
      WRITE(4,*) LAMVAL(L),RHOVAL(W),RELEFF
  830 CONTINUE
C
  840 CONTINUE
C
  850 CONTINUE
C
  860 CONTINUE
      WRITE(4,*) 'CVARX2',COUNTER2,'CRNEG',COUNTER3
C
  900 END
```

Program 3 for a = 5

The following is an example of Program 3 for a = 5, b = 2, and γ = .99. The a, b, and γ value may be changed for the other parameter combinations utilizing Program 3 for a = 5.

```
      INTEGER A,B,N,CLASS,TIME,COUNT,SIZE,PASS,FC,COUNTER2,
         COUNT3,W
      INTEGER NUM,NUM2,COUNTER3
      REAL XMAT(2,5),XVEC(10),SUM,SUMCOL,XDOTI(5),XBAR,TSS2
      REAL WSS,TSS,GAMMA,SMAT(2,5),ZMAT(2,5),GAMVAL(5),
         LAMBDA
      REAL DELTA,DELTAN,DELTA0,CHANGE(5),NXBAR,EPS,X2SSQQ,
         X2PX1
      REAL X2VEC(5),X2MAT(2,5),X1CMAT(2,5),X2SUM,X1CLSMN(3)
      REAL SUM2,CLSSUM(5),Z,X2STD(5),X2SSQ,X2STDSQ,X2STDSUM
      REAL LAMVAL(5),ERVEC(5),EVEC(10),EMAT(2,5),SIGE,SIGR,
         BSS
      REAL X1STD(2,5),X1MSTD(5),BETA0,BETA1,BETA2,
         X1WISTD(2,5),BSS2
      REAL YB(5),YMAT(2,5),RHOVAL(5),RHOHAT,TSSSQ,
         RHOSQ(5),XWPXW
      REAL B1WHAT,B1BHAT,B1HAT1,B1STRHAT,SIGEHAT,
         SIGRHAT,YVEC(10)
      REAL RELEFF,X1PY,X1PR1Y,SUMY,YCLSSUM(5),YBAR,YPR2Y,
         B1HAT2
      REAL YCLSMN(5),XWPIXW,YPXW,YPXWY,YPBETY,YPXB,XBPXB,
         XBPIXB
      REAL YBET(5),YWI(2,5)
      A=5
      B=2
```

```
      N=A*B
C
      EPS = 0.00005
      GAMVAL(1) = .001
      GAMVAL(2) = .2
      GAMVAL(3) = .5
      GAMVAL(4) = .7
      GAMVAL(5) = .99
      LAMVAL(1) = 0
      LAMVAL(2) = .5
      LAMVAL(3) = 1
      LAMVAL(4) = 2
      LAMVAL(5) = 5
      RHOVAL(1) = 0
      RHOVAL(2) = .5
      RHOVAL(3) = 1
      RHOVAL(4) = 2
      RHOVAL(5) = 5
      RHOSQ(1) = SQRT(RHOVAL(1))
      RHOSQ(2) = SQRT(RHOVAL(2))
      RHOSQ(3) = SQRT(RHOVAL(3))
      RHOSQ(4) = SQRT(RHOVAL(4))
      RHOSQ(5) = SQRT(RHOVAL(5))
      X2VEC(1) = 0
      X2VEC(2) = 0
      X2VEC(3) = 0
      X2VEC(4) = 0
      X2VEC(5) = 0
      SIGE = 1
      BETA0 = 0
      BETA1 = 1
C
      WRITE(4,*) 'A',A,'B',B
      COUNT2 = 0
      COUNTER3 = 0
C
      CALL RNOPT(2)
      CALL RNSET(22)
C
      DO 860 M = 1,1000
C
      DO 850 K = 5,5
C
      DO 840 L = 1,5
C
      DO 830 W = 1,5
C
      LOOPD = 0
    5 LOOPC = 0
      LOOPD = LOOPD + 1
      IF (LOOPD.LE.50) THEN
        GO TO 6
      ELSE
        WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
```

```
          GO TO 860
        ENDIF

        DO 250 J = 1,50
C
      6 RNUN(N,XVEC)
        DO 7 I = 1,N
           XVEC(I) = XVEC(I) * 15
      7 CONTINUE
C
        SUM = 0
        COUNT = 1
        DO 20 CLASS = 1,A
              SUMCOL = 0
              DO 10 TIME = 1,B
                  XMAT(TIME,CLASS) = XVEC(COUNT)
                  SUM = SUM + XMAT(TIME,CLASS)
                  SUMCOL = SUMCOL + XMAT(TIME,CLASS)
                  COUNT = COUNT + 1
     10       CONTINUE
              XDOTI(CLASS) = SUMCOL/B
     20 CONTINUE
C
        XBAR = SUM/N
        WSS = 0
        TSS = 0
C
        DO 40 CLASS = 1,A
           DO 30 TIME = 1,B
              SMAT(TIME,CLASS) = XMAT(TIME,CLASS) -
              XDOTI(CLASS)
              WSS = WSS +
              (SMAT(TIME,CLASS))*(SMAT(TIME,CLASS))
              TSS = TSS + (XMAT(TIME,CLASS) - XBAR)*
       +                          (XMAT(TIME,CLASS) - XBAR)
     30    CONTINUE
     40 CONTINUE
C
     50 GAMMA = WSS/TSS
        SUMNEW = 0
C
        IF ((GAMMA.GE.GAMVAL(K)-
           EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
       +   THEN
           GO TO 300
        ENDIF
C
        IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
           SIZE = 0
           FC = 0
           PASS = 1
        ENDIF
C
        IF (GAMMA.GT.GAMVAL(K)+EPS) THEN
```

```
         SIZE = 1
         FC = 0
         PASS = 1
      ENDIF
C
      DELTA = .5
      NXBAR = XBAR
C
   55 LOOPC = LOOPC + 1
      IF (LOOPC.LE.50) THEN
        GO TO 56
      ELSE
        GO TO 5
      ENDIF
   56 WSS = 0
      TSS = 0
      IF ((SIZE.EQ.0).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = .75
         ELSE
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 70 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 60 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
   60       CONTINUE
   70    CONTINUE
         DO 90 CLASS = 1,A
            DO 80 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                             CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
   80          CONTINUE
   90    CONTINUE
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 0
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
```

```
         GO TO 55
C
      ELSE IF ((SIZE.EQ.1).AND.(FC.EQ.0)) THEN
         IF (PASS.EQ.1) THEN
            DELTA0 = 1.5
         ELSE
            DELTA0 = DELTAN
         ENDIF
         PASS = 0
         DELTAN = DELTA0 + DELTA
         NXBAR = DELTAN*XBAR
         DO 110 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 100 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
  100       CONTINUE
  110    CONTINUE
         DO 130 CLASS = 1,A
            DO 120 TIME = 1,B
               WSS = WSS + (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))*
     +                      (ZMAT(TIME,CLASS) -
                            CHANGE(CLASS))
               TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                      (ZMAT(TIME,CLASS) - NXBAR)
  120       CONTINUE
  130    CONTINUE
         GAMMA = WSS/TSS
         IF ((GAMMA.GE.GAMVAL(K)-
            EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +      THEN
            GO TO 300
         ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
            SIZE = 0
            FC = 1
         ELSE
            SIZE = 1
            FC = 0
         ENDIF
         GO TO 55
C
      ELSE IF ((SIZE.EQ.0).AND.(FC.EQ.1)) THEN
         DELTA0 = DELTAN
         DELTA = DELTA*(0.5)
         DELTAN = DELTA0 - DELTA
         NXBAR = DELTAN*XBAR
         DO 150 CLASS = 1,A
            CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
            DO 140 TIME = 1,B
               ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
                  CHANGE(CLASS)
  140       CONTINUE
  150    CONTINUE
```

```
      DO 170 CLASS = 1,A
         DO 160 TIME = 1,B
            WSS = WSS + (ZMAT(TIME,CLASS) -
                         CHANGE(CLASS))*
     +                  (ZMAT(TIME,CLASS) -
                         CHANGE(CLASS))
            TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                  (ZMAT(TIME,CLASS) - NXBAR)
 160     CONTINUE
 170  CONTINUE
      GAMMA = WSS/TSS
      IF ((GAMMA.GE.GAMVAL(K)-
         EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +   THEN
         GO TO 300
      ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
         SIZE = 0
         FC = 1
      ELSE
         SIZE = 1
         FC = 1
      ENDIF
      GO TO 55
C
   ELSE
      DELTA0 = DELTAN
      DELTA = DELTA*(0.5)
      DELTAN = DELTA0 + DELTA
      NXBAR = DELTAN*XBAR
      DO 190 CLASS = 1,A
         CHANGE(CLASS) = DELTAN*XDOTI(CLASS)
         DO 180 TIME = 1,B
            ZMAT(TIME,CLASS) = SMAT(TIME,CLASS) +
               CHANGE(CLASS)
 180     CONTINUE
 190  CONTINUE
      DO 210 CLASS = 1,A
         DO 200 TIME = 1,B
            WSS = WSS + (ZMAT(TIME,CLASS) -
                         CHANGE(CLASS))*
     +                  (ZMAT(TIME,CLASS) -
                         CHANGE(CLASS))
            TSS = TSS + (ZMAT(TIME,CLASS) - NXBAR)*
     +                  (ZMAT(TIME,CLASS) - NXBAR)
 200     CONTINUE
 210  CONTINUE
C
      GAMMA = WSS/TSS
      IF ((GAMMA.GE.GAMVAL(K)-
         EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +   THEN
         GO TO 300
      ELSE IF (GAMMA.LT.GAMVAL(K)-EPS) THEN
         SIZE = 0
```

```
            FC = 1
         ELSE
            SIZE = 1
            FC = 1
         ENDIF
         GO TO 55
C
      ENDIF
C
  250 CONTINUE
C
  300 IF ((GAMMA.GE.GAMVAL(K)-
          EPS).AND.(GAMMA.LE.GAMVAL(K)+EPS))
     +    THEN
          GO TO 305
      ELSE
          WRITE(4,*) 'NO SOLUTION IN 50 LOOPS'
      ENDIF
C
  305 DO 320 CLASS = 1,A
         CLSSUM(CLASS) = 0
         DO 310 TIME = 1,B
            X1CMAT(TIME,CLASS) = ZMAT(TIME,CLASS) - NXBAR
            CLSSUM(CLASS) = CLSSUM(CLASS) +
                            X1CMAT(TIME,CLASS)
  310    CONTINUE
  320 CONTINUE
C
      CLSSUM1 = 0
      CLSSUM2 = 0
      CLSSUM3 = 0
      CLSSUM4 = 0
      CLSSUM5 = 0
      DO 322 CLASS = 1,A
         IF (CLASS.EQ.1) THEN
          CLSSUM1 = CLSSUM(CLASS)
         ELSE IF (CLASS.EQ.2) THEN
          CLSSUM2 = CLSSUM(CLASS)
          ELSE IF (CLASS.EQ.3) THEN
          CLSSUM3 = CLSSUM(CLASS)
          ELSE IF (CLASS.EQ.4) THEN
          CLSSUM4 = CLSSUM(CLASS)
         ELSE
          CLSSUM5 = CLSSUM(CLASS)
         ENDIF
  322  CONTINUE
C
      ZP = (CLSSUM1 - CLSSUM2)**2
      Z2 = (LAMVAL(L)*TSS*SIGE) / (ZP*GAMVAL(K))
      Z3 = SQRT(Z2)
      DO 325 CLASS = 1,A
        IF (CLASS.EQ.1) THEN
        X2VEC(CLASS) = Z3
        ELSE IF (CLASS.EQ.2) THEN
```

```
       X2VEC(CLASS) = -Z3
       ELSE IF (CLASS.EQ.3) THEN
       X2VEC(CLASS) = Z3/2
       ELSE IF (CLASS.EQ.4) THEN
       X2VEC(CLASS) = -Z3/2
       ELSE
       X2VEC(CLASS) = 0
       ENDIF
  325 CONTINUE
C
      SUM2 = 0
      DO 390 CLASS = 1,A
         DO 385 TIME = 1,B
            X2MAT(TIME,CLASS) = X2VEC(CLASS)
  385    CONTINUE
  390 CONTINUE
C
      DO 396 CLASS = 1,A
         X1CLSMN(CLASS) = CLSSUM(CLASS)/B
  396 CONTINUE
C
      BSS = 0
      DO 402 CLASS = 1,A
         DO 401 TIME = 1,B
            BSS = BSS + (X1CLSMN(CLASS)*X1CLSMN(CLASS))
  401    CONTINUE
  402 CONTINUE
C
      X2PX1 = 0
      DO 404 CLASS = 1,A
         DO 403 TIME = 1,B
            X2PX1 = X2PX1
     +          + (X2MAT(TIME,CLASS)*X1CMAT(TIME,CLASS))
  403    CONTINUE
  404 CONTINUE
C
      Z = (SIGE*TSS*TSS*LAMVAL(L))/((X2PX1*X2PX1)*BSS)
      BETA2 = SQRT(Z)
C
      LAMBDA = (BETA2*X2PX1*X2PX1*BETA2*BSS)/(SIGE*TSS*TSS)
C
      X2SSQ = 0
      X2STDSQ = 0
      VARX2STD = 0
      DO 410 CLASS = 1,A
         X2SSQ = X2SSQ + (B*(X2VEC(CLASS)**2))
  410 CONTINUE
C
      X2SSQQ = SQRT(X2SSQ)
      DO 415 CLASS = 1,A
         X2STD(CLASS) = X2VEC(CLASS)/X2SSQQ
         X2STDSQ = X2STDSQ + (B*(X2STD(CLASS)**2))
         VARX2STD = X2STDSQ / (N-1)
  415 CONTINUE
```

```
C
       IF (VARX2STD.GT.1.5) THEN
          COUNTER2 = COUNTER2 + 1
       ELSE
          COUNTER2 = COUNTER2
       ENDIF
C
      CALL RNNOA(A,ERVEC)
      CALL RNNOA(N,EVEC)
      COUNT3 = 1
      DO 435 CLASS = 1,A
         DO 434 TIME = 1,B
            EMAT(TIME,CLASS) = EVEC(COUNT3)
            COUNT3 = COUNT3 + 1
  434    CONTINUE
  435 CONTINUE
  440 CONTINUE
C
      TSSSQ = SQRT(TSS)
      DO 450 CLASS = 1,A
         X1MSTD(CLASS) = (CLSSUM(CLASS)/B)/TSSSQ
  450 CONTINUE
C
      DO 470 CLASS = 1,A
         DO 460 TIME = 1,B
            X1STD(TIME,CLASS) = X1CMAT(TIME,CLASS)/TSSSQ
  460    CONTINUE
  470 CONTINUE
C
      DO 510 CLASS = 1,A
         YB(CLASS) = BETA0 + (BETA1*X1MSTD(CLASS)) +
     +               (BETA2*X2STD(CLASS)) +
     +               (RHOSQ(W)*ERVEC(CLASS))
  510 CONTINUE
C
      DO 540 CLASS = 1,A
         DO 530 TIME = 1,B
            X1WISTD(TIME,CLASS) = X1STD(TIME,CLASS) -
                                  X1MSTD(CLASS)
  530    CONTINUE
  540 CONTINUE
C
      DO 580 CLASS = 1,A
         DO 570 TIME = 1,B
            YMAT(TIME,CLASS) = YB(CLASS) +
            (BETA1*X1WISTD(TIME,CLASS))  SIM03720
     +                            + EMAT(TIME,CLASS)
  570    CONTINUE
  580 CONTINUE
C
      X1PY = 0
      DO 620 CLASS = 1,A
         DO 610 TIME = 1,B
            X1PY = X1PY +
```

```
               (X1STD(TIME,CLASS)*YMAT(TIME,CLASS))
  610 CONTINUE
  620 CONTINUE
C
      B1HAT1 = X1PY
C
      X1PR1Y = 0
      DO 640 CLASS = 1,A
         DO 630 TIME = 1,B
            X1PR1Y = X1PR1Y +
                        (X1MSTD(CLASS)*YMAT(TIME,CLASS))
  630 CONTINUE
  640 CONTINUE
C
      B1BHAT = X1PR1Y/(BSS/TSS)
C
      X1PR2Y = 0
      DO 660 CLASS = 1,A
         DO 650 TIME = 1,B
            X1PR2Y = X1PR2Y +
                     (X1WISTD(TIME,CLASS)*YMAT(TIME,CLASS))
  650 CONTINUE
  660 CONTINUE
C
      B1WHAT = X1PR2Y/(WSS/TSS)
C
      SUMY = 0
      DO 680 CLASS = 1,A
         YCLSSUM(CLASS) = 0
         DO 670 TIME = 1,B
            SUMY = SUMY + YMAT(TIME,CLASS)
            YCLSSUM(CLASS) = YCLSSUM(CLASS) +
               YMAT(TIME,CLASS)
  670    CONTINUE
  680 CONTINUE
C
      DO 685 CLASS = 1,A
         YCLSMN(CLASS) = YCLSSUM(CLASS)/B
  685 CONTINUE
C
      YBAR = SUMY/(A*B)
      YPR2Y = 0
      DO 700 CLASS = 1,A
         DO 690 TIME = 1,B
            YWI(TIME,CLASS) = YMAT(TIME,CLASS) -
               YCLSMN(CLASS)
            YPR2Y = YPR2Y + YWI(TIME,CLASS)*YWI(TIME,CLASS))
  690    CONTINUE
  700 CONTINUE
C
      XWPXW = 0
      YPXW = 0
      DO 720 CLASS = 1,A
         DO 710 TIME = 1,B
```

```
              YPXW = YPXW +
                        (YMAT(TIME,CLASS)*X1WISTD(TIME,CLASS))
              XWPXW = XWPXW +
                     (X1WISTD(TIME,CLASS)*X1WISTD(TIME,CLASS))
              XWPIXW = 1/XWPXW
 710       CONTINUE
 720    CONTINUE
C
        YPXWY = (YPXW*YPXW*XWPIXW)
C
        SIGEHAT = (YPR2Y - YPXWY)/(A*(B-1)-1)
C
        YPBETY = 0
        DO 750 CLASS = 1,A
           YBET(CLASS) = YCLSMN(CLASS) - YBAR
           YPBETY = YPBETY + B*(YBET(CLASS)*YBET(CLASS))
 750    CONTINUE
C
        YPXB = 0
        XBPXB = 0
        DO 770 CLASS = 1,A
           DO 760 TIME = 1,B
              YPXB = YPXB + (YMAT(TIME,CLASS)*X1MSTD(CLASS))
              XBPXB = XBPXB + (X1MSTD(CLASS)*X1MSTD(CLASS))
              XBPIXB = 1/XBPXB
 760       CONTINUE
 770    CONTINUE
C
        YPXBY = YPXB*YPXB*XBPIXB
C
        SIGRHAT = (((YPBETY - YPXBY)/(A-2)) - SIGEHAT)/B
        COUNTER3 = 0
        IF (SIGRHAT.LT.0) THEN
           RHOHAT = 0
           COUNTER3 = COUNTER3 + 1
        ELSE
           RHOHAT = SIGRHAT/SIGEHAT
        ENDIF
C
        D1 = BSS/TSS
        D2 = WSS/TSS
        D3 = 1/((B*SIGRHAT) + SIGEHAT)
        D4 = 1/SIGEHAT
        B1HAT2 = ((D1*B1BHAT) + (D2*B1WHAT))/(D1 + D2)
        B1STRHAT = ((D1*D3*B1BHAT) + (D2*D4*B1WHAT))/((D1*D3)
                    + (D2*D4))
C
        RELEFF = (((1+(B*RHOHAT*GAMVAL(K)))*
                    (1+(B*RHOHAT))*GAMVAL(K))
     +              + LAMVAL(L))/((1+(B*RHOHAT*GAMVAL(K)))**2)
C
        WRITE(4,*) LAMVAL(L),RHOVAL(W),RELEFF
 830    CONTINUE
C
```

```
  840 CONTINUE
C
  850 CONTINUE
C
  860 CONTINUE
      WRITE(4,*)  'CVARX2',COUNTER2,'CRNEG',COUNTER3
C
  900 END
```

APPENDIX F
SAS PROGRAM FOR GENERATING FREQUENCY TABLES

This is a sample program for generating the frequency table, univariate output, and variance information for the parameter combination a = 3, b = 2 and γ = .99. The same program was used for all frequency tables by changing the appropriate values.

```
OPTIONS PS = 50;

DATA SIMA5B3;
  INFILE 'C:\JULIE.SIM';
  INPUT LAMBDA RHO RHOHAT RELEFF;

DATA SIM2; SET SIMA5B3;
  B = 2; GAMMA = .99;
  LIMIT = (1+B*RHO*GAMMA)*(1-GAMMA);
  IF RELEFF <= 1 THEN SIM = 0;
  IF RELEFF > 1 THEN SIM = 1;
  IF LAMBDA <= LIMIT THEN TRUE = 0;
  IF LAMBDA > LIMIT THEN TRUE = 1;
  COUNT4 = SIM + TRUE;
  IF COUNT4 = 0 OR COUNT4 = 2 THEN CORRECT = 1;
  IF COUNT4 = 1 THEN CORRECT = 0;
  TITLE 'A = 3   B = 2   GAMMA = .99';

PROC FREQ DATA = SIM2;
   TABLES LAMBDA*RHO;
   WEIGHT CORRECT;
RUN;

PROC SORT DATA = SIM2;
   BY LAMBDA RHO;
RUN;

DATA SIM3; SET SIM2;
   IF CORRECT = 1 THEN DELETE;
   RELDIFF = RELEFF - 1;
   RELEFF2 = (((1+B*RHO*GAMMA)*(1+B*RHO)*GAMMA)+LAMBDA)/
               ((1+B*RHO*GAMMA)**2);
   RELDIFF2 = RELEFF2 - 1;

PROC UNIVARIATE DATA = SIM3 NORMAL;
   BY LAMBDA RHO;
   VAR RELDIFF;
RUN;

PROC MEANS DATA = SIM2;
   BY LAMBDA RHO;
   VAR RHOHAT;
RUN;
```

```
DATA SIM4; SET SIM2;
   LENGTH COUNTBIG 4;
   IF RHOHAT > 100 THEN COUNTBIG = 1;

PROC FREQ DATA = SIM4;
   TABLES LAMBDA*RHO;
   WEIGHT COUNTBIG;
   TITLE2 'LARGE RHOHAT VALUES';
RUN;
```

APPENDIX G

FREQUENCY AND PERCENTAGE OF CORRECT SELECTION FOR CRITERIA 1

A	B	γ	λ	ρ	Frequency	Percentage
3	2	.001	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	601	60.1
				.5	530	53.0
				1	549	54.9
				2	588	58.8
				5	708	70.8
			2	0	999	99.9
				.5	997	99.7
				1	995	99.5
				2	997	99.7
				5	991	99.1
			5	0	1000	100.0
				.5	1000	100.0
				1	999	99.9
				2	997	99.7
				5	999	99.9
3	2	.2	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	671	67.1
				.5	586	58.6
				1	536	53.6
				2	610	61.0
				5	689	68.9
			2	0	837	83.7
				.5	801	80.1
				1	724	72.4
				2	651	65.1
				5	459	45.9
			5	0	949	94.9
				.5	916	91.6
				1	908	90.8
				2	840	84.0
				5	732	73.2

A	B	γ	λ	ρ	Frequency	Percentage
3	2	.5	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	701	70.1
				.5	606	60.6
				1	435	43.5
				2	543	54.3
				5	667	66.7
			2	0	836	83.6
				.5	770	77.0
				1	690	69.0
				2	624	62.4
				5	533	53.3
			5	0	945	94.5
				.5	899	89.9
				1	852	85.2
				2	791	79.1
				5	679	67.9
3	2	.7	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	628	62.8
				.5	470	47.0
				1	530	53.0
				2	629	62.9
				5	711	71.1
			1	0	790	79.0
				.5	691	69.1
				1	627	62.7
				2	458	45.8
				5	624	62.4
			2	0	881	88.1
				.5	813	81.3
				1	726	72.6
				2	667	66.7
				5	471	47.1
			5	0	918	91.8
				.5	910	91.0
				1	879	87.9
				2	833	83.3
				5	730	73.0

A	B	γ	λ	ρ	Frequency	Percentage
3	2	.99	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	980	98.0
				.5	939	93.9
				1	941	94.1
				2	906	90.6
				5	848	84.8
			1	0	981	98.1
				.5	977	97.7
				1	968	96.8
				2	937	93.7
				5	896	89.6
			2	0	991	99.1
				.5	993	99.3
				1	989	98.9
				2	967	96.7
				5	945	94.5
			5	0	998	99.8
				.5	991	99.1
				1	989	98.9
				2	987	98.7
				5	967	96.7
3	3	.001	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	722	72.2
				.5	416	41.6
				1	522	52.2
				2	620	62.0
				5	739	73.9
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	999	99.9
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0

A	B	γ	λ	ρ	Frequency	Percentage
3	3	.2	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	745	74.5
				.5	442	44.2
				1	516	51.6
				2	600	60.0
				5	720	72.0
			2	0	940	94.0
				.5	836	83.6
				1	759	75.9
				2	691	69.1
				5	488	48.8
			5	0	993	99.3
				.5	975	97.5
				1	940	94.0
				2	889	88.9
				5	742	74.2
3	3	.5	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	773	77.3
				.5	638	63.8
				1	458	45.8
				2	552	55.2
				5	687	68.7
			2	0	901	90.1
				.5	809	80.9
				1	733	73.3
				2	383	38.3
				5	558	55.8
			5	0	986	98.6
				.5	946	94.6
				1	906	90.6
				2	831	83.1
				5	662	66.2

A	B	γ	λ	ρ	Frequency	Percentage
3	3	.7	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	715	71.5
				.5	451	45.1
				1	561	56.1
				2	638	63.8
				5	748	74.8
			1	0	867	86.7
				.5	699	69.9
				1	639	63.9
				2	491	49.1
				5	657	65.7
			2	0	938	93.8
				.5	856	85.6
				1	790	79.0
				2	674	67.4
				5	503	50.3
			5	0	991	99.1
				.5	961	96.1
				1	929	92.9
				2	860	86.0
				5	694	69.4
3	3	.99	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	995	99.5
				1	982	98.2
				2	950	95.0
				5	870	87.0
			1	0	1000	100.0
				.5	999	99.9
				1	995	99.5
				2	978	97.8
				5	942	94.2
			2	0	999	99.9
				.5	1000	100.0
				1	999	99.9
				2	995	99.5
				5	988	98.8
			5	0	1000	100.0
				.5	1000	100.0
				1	999	99.9
				2	1000	100.0
				5	997	99.7

A	B	γ	λ	ρ	Frequency	Percentage
3	6	.001	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	739	73.9
				.5	523	52.3
				1	618	61.8
				2	724	72.4
				5	799	79.9
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
3	6	.2	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	729	72.9
				.5	505	50.5
				1	601	60.1
				2	692	69.2
				5	791	79.1
			2	0	970	97.0
				.5	825	82.5
				1	695	69.5
				2	458	45.8
				5	619	61.9
			5	0	999	99.9
				.5	971	97.1
				1	913	91.3
				2	810	81.0
				5	365	36.5

A	B	γ	λ	ρ	Frequency	Percentage
3	6	.5	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	816	81.6
				.5	418	41.8
				1	553	55.3
				2	661	66.1
				5	738	73.8
			2	0	952	95.2
				.5	750	75.0
				1	332	33.2
				2	475	47.5
				5	647	64.7
			5	0	997	99.7
				.5	945	94.5
				1	870	87.0
				2	765	76.5
				5	460	46.0
3	6	.7	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	695	69.5
				.5	521	52.1
				1	611	61.1
				2	718	71.8
				5	817	81.7
			1	0	892	89.2
				.5	663	66.3
				1	461	46.1
				2	580	58.0
				5	707	70.7
			2	0	973	97.3
				.5	819	81.9
				1	707	70.7
				2	414	41.4
				5	593	59.3
			5	0	997	99.7
				.5	963	96.3
				1	906	90.6
				2	789	78.9
				5	403	40.3

A	B	γ	λ	ρ	Frequency	Percentage
3	6	.99	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	996	99.6
				1	983	98.3
				2	914	91.4
				5	757	75.7
			1	0	1000	100.0
				.5	1000	100.0
				1	995	99.5
				2	979	97.9
				5	923	92.3
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	998	99.8
				5	984	98.4
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
3	12	.001	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	719	71.9
				.5	618	61.8
				1	689	68.9
				2	778	77.8
				5	863	86.3
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	999	99.9
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0

A	B	γ	λ	ρ	Frequency	Percentage
3	12	.2	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	780	78.0
				.5	581	58.1
				1	691	69.1
				2	783	78.3
				5	834	83.4
			2	0	983	98.3
				.5	705	70.5
				1	435	43.5
				2	570	57.0
				5	702	70.2
			5	0	999	99.9
				.5	939	93.9
				1	823	82.3
				2	674	67.4
				5	516	51.6
3	12	.5	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	833	83.3
				.5	551	55.1
				1	651	65.1
				2	745	74.5
				5	810	81.0
			2	0	973	97.3
				.5	330	33.0
				1	476	47.6
				2	611	61.1
				5	739	73.9
			5	0	999	99.9
				.5	873	87.3
				1	740	74.0
				2	385	38.5
				5	568	56.8

A	B	γ	λ	ρ	Frequency	Percentage
3	12	.7	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	740	74.0
				.5	610	61.0
				1	697	69.7
				2	784	78.4
				5	883	88.3
			1	0	903	90.3
				.5	443	44.3
				1	577	57.7
				2	674	67.4
				5	816	81.6
			2	0	989	98.9
				.5	736	73.6
				1	398	39.8
				2	556	55.6
				5	725	72.5
			5	0	1000	100.0
				.5	935	93.5
				1	795	79.5
				2	334	33.4
				5	527	52.7
3	12	.99	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	988	98.8
				1	942	94.2
				2	840	84.0
				5	376	37.6
			1	0	1000	100.0
				.5	999	99.9
				1	991	99.1
				2	943	94.3
				5	790	79.0
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	993	99.3
				5	898	89.8
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	992	99.2

A	B	γ	λ	ρ	Frequency	Percentage
5	2	.001	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	707	70.7
				.5	485	48.5
				1	613	61.3
				2	794	79.4
				5	905	90.5
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
5	2	.2	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	728	72.8
				.5	505	50.5
				1	606	60.6
				2	780	78.0
				5	885	88.5
			2	0	954	95.4
				.5	893	89.3
				1	803	80.3
				2	672	67.2
				5	584	58.4
			5	0	997	99.7
				.5	980	98.0
				1	968	96.8
				2	930	93.0
				5	794	79.4

A	B	γ	λ	ρ	Frequency	Percentage
5	2	.5	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	789	78.9
				.5	632	63.2
				1	494	49.4
				2	645	64.5
				5	860	86.0
			2	0	930	93.0
				.5	854	85.4
				1	766	76.6
				2	620	62.0
				5	618	61.8
			5	0	978	97.8
				.5	956	95.6
				1	924	92.4
				2	891	89.1
				5	697	69.7
5	2	.7	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	683	68.3
				.5	530	53.0
				1	661	66.1
				2	762	76.2
				5	903	90.3
			1	0	885	88.5
				.5	737	73.7
				1	638	63.8
				2	550	55.0
				5	776	77.6
			2	0	962	96.2
				.5	878	87.8
				1	817	81.7
				2	674	67.4
				5	551	55.1
			5	0	993	99.3
				.5	979	97.9
				1	964	96.4
				2	903	90.3
				5	759	75.9

A	B	γ	λ	ρ	Frequency	Percentage
5	2	.99	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	999	99.9
				.5	999	99.9
				1	988	98.8
				2	978	97.8
				5	898	89.8
			1	0	998	99.8
				.5	998	99.8
				1	999	99.9
				2	994	99.4
				5	954	95.4
			2	0	1000	100.0
				.5	999	99.9
				1	999	99.9
				2	997	99.7
				5	994	99.4
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
5	3	.001	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	734	73.4
				.5	555	55.5
				1	718	71.8
				2	840	84.0
				5	942	94.2
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0

A	B	γ	λ	ρ	Frequency	Percentage
5	3	.2	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	805	80.5
				.5	531	53.1
				1	693	69.3
				2	842	84.2
				5	936	93.6
			2	0	990	99.0
				.5	912	91.2
				1	797	79.7
				2	653	65.3
				5	668	66.8
			5	0	999	99.9
				.5	995	99.5
				1	986	98.6
				2	950	95.0
				5	766	76.6
5	3	.5	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	866	86.6
				.5	568	56.8
				1	572	57.2
				2	747	74.7
				5	917	91.7
			2	0	991	99.1
				.5	881	88.1
				1	760	76.0
				2	456	45.6
				5	704	70.4
			5	0	999	99.9
				.5	994	99.4
				1	968	96.8
				2	881	88.1
				5	622	62.2

A	B	γ	λ	ρ	Frequency	Percentage
5	3	.7	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	714	71.4
				.5	567	56.7
				1	711	71.1
				2	855	85.5
				5	941	94.1
			1	0	937	93.7
				.5	749	74.9
				1	567	56.7
				2	619	61.9
				5	847	84.7
			2	0	996	99.6
				.5	926	92.6
				1	857	85.7
				2	667	66.7
				5	655	65.5
			5	0	999	99.9
				.5	996	99.6
				1	986	98.6
				2	923	92.3
				5	702	70.2
5	3	.99	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	999	99.9
				2	992	99.2
				5	898	89.8
			1	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	991	99.1
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	996	99.6
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0

A	B	γ	λ	ρ	Frequency	Percentage
5	6	.001	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	776	77.6
				.5	729	72.9
				1	838	83.8
				2	940	94.0
				5	970	97.0
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
5	6	.2	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	822	82.2
				.5	656	65.6
				1	816	81.6
				2	916	91.6
				5	977	97.7
			2	0	997	99.7
				.5	860	86.0
				1	651	65.1
				2	590	59.0
				5	873	87.3
			5	0	1000	100.0
				.5	998	99.8
				1	982	98.2
				2	866	86.6
				5	473	47.3

A	B	γ	λ	ρ	Frequency	Percentage
5	6	.5	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	899	89.9
				.5	563	56.3
				1	753	75.3
				2	898	89.8
				5	973	97.3
			2	0	997	99.7
				.5	774	77.4
				1	421	42.1
				2	680	68.0
				5	872	87.2
			5	0	1000	100.0
				.5	992	99.2
				1	933	93.3
				2	764	76.4
				5	631	63.1
5	6	.7	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	737	73.7
				.5	706	70.6
				1	839	83.9
				2	929	92.9
				5	982	98.2
			1	0	968	96.8
				.5	587	58.7
				1	635	63.5
				2	777	77.7
				5	942	94.2
			2	0	999	99.9
				.5	898	89.8
				1	638	63.8
				2	559	55.9
				5	814	81.4
			5	0	1000	100.0
				.5	992	99.2
				1	965	96.5
				2	819	81.9
				5	542	54.2

A	B	γ	λ	ρ	Frequency	Percentage
5	6	.99	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	997	99.7
				2	982	98.2
				5	795	79.5
			1	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	957	95.7
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	999	99.9
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
5	12	.001	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	847	84.7
				.5	827	82.7
				1	911	91.1
				2	978	97.8
				5	994	99.4
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0

A	B	γ	λ	ρ	Frequency	Percentage
5	12	.2	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	850	85.0
				.5	799	79.9
				1	904	90.4
				2	961	96.1
				5	989	98.9
			2	0	1000	100.0
				.5	695	69.5
				1	586	58.6
				2	780	78.0
				5	929	92.9
			5	0	1000	100.0
				.5	984	98.4
				1	891	89.1
				2	653	65.3
				5	709	70.9
5	12	.5	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			1	0	909	90.9
				.5	742	74.2
				1	870	87.0
				2	951	95.1
				5	983	98.3
			2	0	1000	100.0
				.5	407	40.7
				1	647	64.7
				2	833	83.3
				5	938	93.8
			5	0	1000	100.0
				.5	931	93.1
				1	755	75.5
				2	519	51.9
				5	820	82.0

A	B	γ	λ	ρ	Frequency	Percentage
5	12	.7	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	766	76.6
				.5	849	84.9
				1	930	93.0
				2	981	98.1
				5	992	99.2
			1	0	975	97.5
				.5	638	63.8
				1	795	79.5
				2	916	91.6
				5	971	97.1
			2	0	1000	100.0
				.5	690	69.0
				1	552	55.2
				2	770	77.0
				5	945	94.5
			5	0	1000	100.0
				.5	973	97.3
				1	851	85.1
				2	435	43.5
				5	772	77.2
5	12	.99	0	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0
			.5	0	1000	100.0
				.5	998	99.8
				1	985	98.5
				2	864	86.4
				5	462	46.2
			1	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	994	99.4
				5	812	81.2
			2	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	966	96.6
			5	0	1000	100.0
				.5	1000	100.0
				1	1000	100.0
				2	1000	100.0
				5	1000	100.0

2

VITA

Julia Kay Sawyer

Candidate for the Degree of

Doctor of Philosophy

Thesis: THE EFFECT OF BIAS ON LEAST SQUARES ESTIMATORS OF THE SLOPE PARAMETER

Major Field: Statistics

Biographical:

Personal Data: Born in Ottumwa, Iowa, on October 6, 1960, the daughter of Donald W. and Shirley K. Westfall.

Education: Graduated from Broken Arrow High School, Broken Arrow, Oklahoma in May 1979; received Bachelor of Science degree in Psychology from Northwestern Oklahoma State University, Alva, Oklahoma in May 1991; received Master of Science degree in Statistics from Oklahoma State University, Stillwater, Oklahoma in December 1993. Completed the requirements for the Doctor of Philosophy degree with a major in Statistics at Oklahoma State University in May 1996.

Experience: Employed by Oklahoma State University, Department of Statistics as graduate teaching associate, 1991 to present.