1989

Identifying Outliers in a Random Effects Model For Longitudinal Data

Tamarah Crouse Dishman

University of North Florida
INDENTIFYING OUTLIERS IN A
RANDOM EFFECTS MODEL FOR LONGITUDINAL DATA

by

Tamarah Crouse Dishman

A thesis submitted to the Department of Mathematics and Statistics in partial fulfillment of the requirements for the degree of

Master of Arts in Mathematical Sciences

University of North Florida
College of Arts and Sciences

December, 1989
The thesis of Tamarah Crouse Dishman is approved:

Signature deleted  12/14/89

Signature deleted  12/14/89

Signature deleted  12/14/89

Committee Chairperson

Accepted for The Department:

Signature deleted  12/14/89

Chairperson

Accepted for the College:

Signature deleted  12/14/89

Dean

Accepted for the University:

Signature deleted  12/14/89

Interim Vice-President for Academic Affairs
I extend my sincere appreciation to Graduate Director Dr. Donna Mohr, the advisor of this project, for her invaluable guidance and support. I am also grateful to the Faculty, Staff and Students of this University for their interest and contributions.

I also wish to thank my parents for encouraging and nurturing my scholastic endeavors and especially to my husband for his loving support of my professional goals.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acknowledgements</td>
<td>(iii)</td>
</tr>
<tr>
<td>Abstract</td>
<td>(vi)</td>
</tr>
<tr>
<td>Chapter 1 - Introduction</td>
<td></td>
</tr>
<tr>
<td>Section 1 - Random effects model for longitudinal data</td>
<td>2</td>
</tr>
<tr>
<td>Section 2 - Estimation of parameters</td>
<td>3</td>
</tr>
<tr>
<td>Chapter 2 - Method of Identifying Non-Trackers</td>
<td></td>
</tr>
<tr>
<td>Section 1 - Method of Identification</td>
<td>11</td>
</tr>
<tr>
<td>Section 2 - Explanation of computer algorithm</td>
<td>14</td>
</tr>
<tr>
<td>Chapter 3 - Conclusion</td>
<td>20</td>
</tr>
<tr>
<td>Appendix 1</td>
<td>27</td>
</tr>
<tr>
<td>Appendix 2</td>
<td>29</td>
</tr>
<tr>
<td>Appendix 3</td>
<td>30</td>
</tr>
<tr>
<td>References</td>
<td>46</td>
</tr>
<tr>
<td>Vita</td>
<td>48</td>
</tr>
</tbody>
</table>
List of Tables and Figures

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>Parameters for Trackers</td>
<td>15</td>
</tr>
<tr>
<td>Table 2</td>
<td>Simulations Run with only Trackers Present</td>
<td>16</td>
</tr>
<tr>
<td>Figure 1</td>
<td>Graph of Expected Values for Non-Trackers</td>
<td>17</td>
</tr>
<tr>
<td>Table 3</td>
<td>Simulations Run with Non-Trackers Present</td>
<td>18</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Outline for Computer Algorithm</td>
<td>19</td>
</tr>
<tr>
<td>Table 4</td>
<td>Results with Trackers Only</td>
<td>23</td>
</tr>
<tr>
<td>Table 5</td>
<td>Parameter Estimates with Trackers Only</td>
<td>24</td>
</tr>
<tr>
<td>Table 6</td>
<td>Results with Non-Trackers Present</td>
<td>25</td>
</tr>
<tr>
<td>Table 7</td>
<td>Parameter Estimates with Non-Trackers Present</td>
<td>26</td>
</tr>
</tbody>
</table>
Identifying non-tracking individuals in a population of longitudinal data has many applications as well as complications. The analysis of longitudinal data is a special study in itself. There are several accepted methods, of those we chose a two-stage random effects model coupled with the Estimation Maximization Algorithm (E-M Algorithm). Our project consisted of first estimating population parameters using the previously mentioned methods. The Mahalanobis distance was then used to sequentially identify and eliminate non-trackers from the population. Computer simulations were run in order to measure the algorithm’s effectiveness.

Our results show that the average specificity for the repetitions for each simulation remained at the 99% level. The sensitivity was best when only a single non-tracker was present with a very different parameter $a$. The sensitivity of the program decreased when more than one tracker was present, indicating our method of identifying a non-tracker is not effective when the estimates of the population parameters are contaminated.
Chapter 1 - Introduction

According to Ware (1984) longitudinal studies can be loosely defined as studies in which the response of each individual is observed on two or more occasions. There are obviously many applications of longitudinal studies in the medical and social fields. The objectives of studies of this type are to characterize patterns of response and change over time. This motivates the definition of tracking given by Ware and Wu (1981) as the prediction of future values based on repeated measurements of the same characteristic obtained over time for each of a cohort chart of individuals. In this thesis, non-trackers will be defined as individuals whose longitudinal observations do not seem to belong to the same distribution as the rest of the tracking population.

In the remainder of this chapter a popular model for analyzing longitudinal data called the random effects model (Laird and Ware, 1982) will be introduced and explained. The derivations of the equations from Diem and Liukkonen (1988) for fitting the model will be given in detail. Chapter 2 will include the criteria for distinguishing trackers from non-trackers and conclude with a description of the computer simulation of the method. The computer program will be tested for its specificity (defined as its
behavior when no non-trackers are present) as well as its sensitivity (measured by its ability to detect non-trackers when they are present). Results of the simulations and overall conclusions appear in Chapter 3.

Section 1: Random effects model for longitudinal data

Laird and Ware (1982) introduced a two stage model for the analysis of the highly unbalanced data sets obtained from longitudinal studies. In the first stage, the distribution of the characteristics being measured has the same form for each individual, but the parameters vary over individuals. The second stage describes the distribution of these individual parameters or random effects.

Stage 1: for each unit $i$

$$y_i = X_i \alpha + Z_i b_i + e_i$$

where $y_i$ is the vector of $n_i$ observations from individual $i$, $\alpha$ is a $p \times 1$ vector of the unknown population parameters, $X_i$ is a known design matrix linking $\alpha$ to $y_i$ for each individual, $b_i$ is the $k \times 1$ vector of individual effects and $Z_i$ is the known design matrix linking $b_i$ to $y_i$ for each individual. The $e_i$ vectors are distributed $N(0,R_i)$ and assumed to be independent while $\alpha$ is considered fixed and $b_i$ is a random vector as described in stage 2. Throughout the rest of our work we take $R_i = \sigma^2 I$. 

Stage 2

The $b_i$ are distributed as $N(0,D)$, independently of each other and of the $e_i$. $D$ is a $k \times k$ positive definite covariance matrix. The population parameters, $\alpha$, are treated as fixed effects.

The $y_i$ are independent and distributed $N(X_i \alpha, Z_i D Z_i^T + \sigma^2 I)$. The main disadvantage of this model is the strong assumption made about the structure of the covariance matrix of the $y_i$ given above.

Section 2: Estimation of parameters

In this section, equations for estimating $\alpha, \sigma^2$ and $D$ will be developed. Since there are no closed form solutions we will derive the iterative solutions from maximum likelihood estimates using the Estimation Maximization Algorithm (comprised of E-step and M-step and denoted E-M Algorithm) given by Dempster et al (1977). We apply the E-M Algorithm to the random effects model following Diem and Liukkonen (1988). The derivations omitted by them are included in this paper as well as the equations. The idea behind the E-M Algorithm is very simple:

1. In the E-step, the $b_i$ are treated as missing values and are replaced by estimates of $\hat{b}_i$. This estimate is calculated using current estimates of $\alpha, \sigma^2, D$.

2. In the M-step, parameters $\alpha, \sigma^2$ and $D$ are estimated using the $y_i$ and $b_i$.

The algorithm is repeated until convergence is obtained or the maximum allowed iterations is reached. The
derivation of the equations is as follows.

**E-Step**

First note that the joint probability distribution for \( y_i \) and \( b_i \) given \( \theta=(\alpha, \sigma^2, D) \) is given by:

\[
f(y_i, b_i | \theta) = f(y_i | b_i, \theta) \cdot f(b_i | \theta)
\]

\[
= c_1 \cdot \frac{1}{\text{det} |\sigma^2 I|^{1/2}} \exp \left\{ -\frac{1}{2\sigma^2} (y_i - x_i' \alpha - z_i' b_i)' (y_i - x_i' \alpha - z_i' b_i) \right\} \\
\cdot \frac{1}{\text{det} |D|^{1/2}} \exp \left\{ -\frac{1}{2} (b_i' D^{-1} b_i) \right\}
\]

where \( c_1 \) is a constant

**NOTE:**

\[
(y_i - x_i' \alpha - z_i' b_i)' (y_i - x_i' \alpha - z_i' b_i) = [(y_i - x_i' \alpha)' - (z_i' b_i)' ][(y_i - x_i' \alpha) - (z_i b_i)]
\]

\[
= (y_i - x_i' \alpha)' (y_i - x_i' \alpha) - 2(y_i - x_i' \alpha)' z_i b_i + b_i' z_i' z_i b_i
\]

Now what we need is the conditional pdf of \( b_i \) given \( \theta \)

\[
f(b_i | y_i, \theta) = \frac{f(b_i, y_i | \theta)}{f(y_i | \theta)}
\]

Note that the denominator above is a constant with respect to \( b_i \). We collect all of the \( b_i \) terms in \( f(b_i, y_i | \theta) \) and let the remaining terms become one constant, \( C_2 \).

Therefore,
\( f(b_i | y_i, \theta) = c_2 \exp \left\{ \frac{1}{\sigma^2} (y_i - x_i \alpha)^T z_i b_i \right\} \)

\[ \cdot \exp \left\{ - \frac{1}{2\sigma^2} (b_i^T z_i z_i b_i) - \frac{1}{2} (b_i^T D^{-1} b_i) \right\} \]

\[ = c_2 \exp \left\{ \frac{1}{2\sigma^2} \left[ 2(y_i - x_i \alpha)^T z_i b_i - b_i^T (z_i^T z_i + D^{-1} \sigma^2) b_i \right] \right\} \]

\[ = c_2 \exp \left\{ \frac{-1}{2} \left[ \right. \right. \]

\[ \left. \left. \frac{-2(y_i - x_i \alpha)^T z_i b_i + b_i^T \left( z_i^T z_i + D^{-1} \sigma^2 \right) b_i \right] \right\} \]

\[ = c_2 \exp \left\{ \frac{-1}{2} \right. \left[ b_i^T A b_i - 2 \left( y_i - x_i \alpha \right)^T z_i b_i / \sigma^2 \right] \right\} \]

where \( A = (z_i^T z_i + D^{-1} \sigma^2) / \sigma^2 \). This can be recognized as the general form of the multivariate normal distribution. The variance is found directly by

\[ \text{Var}(b_i | y_i, \theta) = A^{-1} = \left( \frac{z_i^T z_i + D^{-1} \sigma^2}{\sigma^2} \right)^{-1} = \sigma^2 \left( z_i^T z_i + D^{-1} \sigma^2 \right)^{-1} \]

From Appendix 1, it follows that:
\[ E(b_i | y_i, \theta) = (z_i^T z_i + D^{-1} \sigma^2)^{-1} z_i^T (y_i - x_i \alpha) = \hat{b}_i \] 

**M-Step**

All sums below are over \( i = 1, m \).

Using the values \( \hat{b}_i \) calculated in the E-step we want to maximize \( H(\theta) = E(\ln[f(y_i, b_i | \theta)] | y_i, \theta) \), ignoring constants it follows that:

\[
\begin{align*}
E & \left\{ \sum_{i=1}^{m} \left[ \frac{-n_i}{2} \ln(\sigma^2) - \frac{1}{2\sigma^2} \left( y_i - x_i \alpha - z_i b_i \right)^T \left( y_i - x_i \alpha - z_i b_i \right) - \frac{1}{2} \ln(\det D) \right. \right. \\
& \quad \left. \left. - \frac{1}{2} \left( b_i^T D^{-1} b_i \right) \right] \right\} = \\
& \sum_{i=1}^{m} \left[ \frac{-n_i}{2} \ln(\sigma^2) - \frac{1}{2\sigma^2} \left( y_i - x_i \alpha \right)^T \left( y_i - x_i \alpha \right) + \frac{1}{\sigma^2} \left( y_i - x_i \alpha \right)^T z_i E\left( b_i | y_i, \theta \right) \right. \\
& \quad \left. - \frac{1}{2\sigma^2} E\left( b_i^T z_i^T z_i b_i | y_i, \theta \right) - \frac{1}{2} \ln(\det D) - \frac{1}{2} E\left( b_i^T D^{-1} b_i | y_i, \theta \right) \right]\end{align*}
\]
Note: for the above equation \( N = \sum n_i \)

Now, we use \( H(\theta) \) to derive expressions for \( \hat{\alpha}, \hat{\sigma}^2 \) and \( \hat{D} \).

By differentiating \( H(\theta) \) with respect to each variable, setting the expression equal to zero and solving for the given variable, a maximum is obtained.
First consider $\hat{\alpha}$:

Note:

\[-\frac{1}{2\sigma^2}\sum_{i=1}^{m}(y_i-x_i\alpha)^T(y_i-x_i\alpha) = -\frac{1}{2\sigma^2}\sum_{i=1}^{m}y_i^Ty_i + \frac{1}{\sigma^2}\sum_{i=1}^{m}y_i^Tx_i\alpha + \frac{1}{2\sigma^2}\sum_{i=1}^{m}x_i^Tx_i\alpha\]

and,

\[-\frac{1}{\sigma}\sum_{i=1}^{m}(y_i-x_i\alpha)^Tz_i\hat{\beta}_i = -\frac{1}{\sigma}\sum_{i=1}^{m}y_i^Tz_i\hat{\beta}_i + \frac{1}{\sigma}\sum_{i=1}^{m}x_i^Tz_i\hat{\beta}_i\]

therefore,

\[\frac{\partial H(\theta)}{\partial \alpha} = -\frac{1}{\sigma}\sum_{i=1}^{m}x_i^Ty_i - \frac{1}{\sigma}\sum_{i=1}^{m}x_i^Tx_i\alpha - \frac{1}{\sigma}\sum_{i=1}^{m}x_i^Tz_i\hat{\beta}_i = 0\]

it follows that,

\[\sum_{i=1}^{m}x_i^T x_i\alpha = \sum_{i=1}^{m}x_i^Ty_i - \sum_{i=1}^{m}x_i^Tz_i\hat{\beta}_i = \sum_{i=1}^{m}x_i^T(y_i-z_i\hat{\beta}_i)\]
and,

\[ \hat{\alpha} = \left( \sum_{i=1}^{m} x_i^T x_i \right)^{-1} \sum_{i=1}^{m} x_i^T \left( y_i - z_i \hat{b}_i \right) \]  

(3)

Now consider \( \sigma^2 \):

\[ \frac{\partial H(\theta)}{\partial \sigma^2} = \frac{-N}{2\sigma^2} + \frac{m}{2\sigma^4} \left[ \sum_{i=1}^{m} (y_i - x_i \alpha)^T \left( y_i - x_i \alpha \right) \right] \left( \sum_{i=1}^{m} (y_i - x_i \alpha)^T z_i \hat{b}_i \right) \]

\[ + \frac{m}{2\sigma^4} \text{tr} \left[ z_i^T z_i \sqrt{\text{V}(b_i | y_i, \theta)} \right] + \frac{m}{2\sigma^4} \sum_{i=1}^{m} b_i^T z_i z_i \hat{b}_i = 0 \]

it follows that,

\[ \sigma^2 N - \sum_{i=1}^{m} (y_i - x_i \alpha)^T (y_i - x_i \alpha) + 2 \sum_{i=1}^{m} (y_i - x_i \alpha)^T z_i \hat{b}_i \]

\[ - \sum_{i=1}^{m} \text{tr} \left[ z_i^T z_i \sqrt{\text{V}(b_i | y_i, \theta)} \right] - \sum_{i=1}^{m} \hat{b}_i^T z_i z_i \hat{b}_i = 0 \]
therefore,

\[
\hat{\sigma}^2 = \frac{1}{N} \sum \{(y_i - x_i \alpha)^T (y_i - x_i \alpha) - 2(y_i - x_i \alpha)^T z_i \hat{b}_i \\
+ \text{tr}[z_i^T z_i \nu(b_i | y_i, \theta)] + \hat{b}_i^T z_i^T z_i \hat{b}_i\} \tag{4}
\]

Finally consider \(\hat{b}\):

In appendix 2 we present some facts about partial derivatives with respect to \(D\). Using those facts we can show,

\[
\frac{\partial H(\theta)}{\partial D} = \frac{-m}{2} D^{-1} + \frac{1}{2} D^{-1} \sum \hat{b}_i \hat{b}_i^T D^{-1} + \frac{1}{2} D^{-1} \sum \nu(b_i | y_i, \theta) D^{-1} = 0
\]

it follows that,

\[-mD + \sum \hat{b}_i \hat{b}_i^T + \sum \nu(b_i | y_i, \theta) = 0
\]

\[\Rightarrow \hat{b} = \frac{1}{m} \sum \hat{b}_i \hat{b}_i^T + \nu(b_i | y_i, \theta) \] \tag{5}

We have now verified the equations given by Diem and Liukkonen (1988).
Chapter 2 - Method of Identifying Non-Trackers

In order to test the method discussed in Chapter 1, we used the equations 1-5 and implemented them in a computer program. In order to make computation easier, only the balanced case was addressed. What follows is an explanation of the criterion used in the algorithm for identifying non-trackers, a flow chart of the program and a list of the various simulations that were run.

Section 1: Method of Identification

As mentioned in the introduction, non-trackers will be identified as those individuals whose observations do not seem to belong to the distribution of the tracking population. The criterion we have selected to make this determination is called the Mahalanobis distance and is defined as follows:

for each individual i,

\[ D_i = (y_i - \mu)^T (\text{var } y)^{-1} (y_i - \mu) \]

\[ = (y_i - x_i \hat{\alpha})^T (Z_i \hat{D} Z_i^T + \sigma^2 I)^{-1} (y_i - x_i \hat{\alpha}) \]  

(6)

since we assume that each individual is normally distributed with mean \( x_i \alpha \) and variance \( Z_i \hat{D} Z_i^T + \sigma^2 I \). If \( \alpha, \sigma^2 \) and \( D \) were known and used in place of their estimates in equation
(6), clearly, $D_1$ would have a chi-square distribution with $n$ degrees of freedom where $n$ is the dimension of $y$.

In order to "weed out" non-trackers, we will first find the individual with the largest Mahalanobis distance. The p-value is calculated for that individual and compared to a previously determined significance level (denoted "signif"). If the p-value is less than the significance level, the individual is considered a non-tracker and eliminated from the tracking population. New population parameters are calculated and the process is repeated until the p-value of the maximum $D_1$ in the current iteration is not less than the significance level. At that time the parameters of the tracking population are given as well as the number of non-trackers.

Due to our approximation of the $D_1$'s being independently distributed as chi-square each with $n$ degrees of freedom, we arrived at our calculation of the p-value by using order statistics. Each time through the "weeding out" process we are interested in the individual with the maximum $D_1$. Note that from equation 6, this is a measurement of the observation with the maximum distance from the normal distribution with parameters calculated from equations 1-5. Examination of the probability distribution of the maximum order statistic for this type of distribution leads to a p-value expressed as

$$p\text{-value}=1-F(d_{\text{max}})^m$$
where $d_{\text{max}}$ stands for maximum $D_i$ from $X^2(n)$ and $F(d_{\text{max}})$ equals the cumulative distribution function. The program for computing $F(d_{\text{max}})$ is taken from Press et al (1986).

In order to test the sensitivity (probability that an individual is identified as a non-tracker given that they are really a non-tracker) and specificity (probability that an individual is identified as a tracker given that they really are a tracker) of our algorithm, several simulations will be run. Combinations of the number of non-trackers present, the number of individuals, the number of observations per individual, the magnitude of $\sigma^2$, significance levels and values used for $X$, $Z$ and $a$ are listed in tables at the end of this chapter. The results of the simulations described above appear in Chapter 3.

In an attempt to clarify the relationship between trackers, non-trackers and their parameters a pictorial representation appears in Figure 1 at the end of this chapter. Trackers and non-trackers are sketched on the same axis with respect to $Xa$, their expected values, for $n=5$ and $n=10$. Both case A and case B for non-trackers are shown. In case A, $a=(5,-4,2)$, the opposite slopes for the sketches indicate that the non-trackers are drastically different from the trackers. For case B, $a=(7,0,1)$, there is only a slight difference between the non-trackers and trackers. Given this information, we would expect case A type of non-tracker to be easier to identify.
**Section 2: Explanation of Computer Algorithm**

As sketched in Figure 2, the calling program is RANCOEF2. It begins by getting parameters for trackers and non-trackers after which it generates $y_i$ for each. It then calls the subroutine FITRCB2 which is designed to first make initial estimates for the population parameters, then improve these estimates using the E-M algorithm. Next, the subroutine FIND is called to "weed out" the non-trackers. If any individual is eliminated FITRCB2 is called to recalculate the parameter estimates of the tracking population then FIND is called again. After all non-trackers have been "weeded out" we return to the main program where parameter information is recorded. There are 50 repetitions of each simulation and overall statistics for each type of simulation are calculated and given in Tables 4 and 6 of Chapter 3. The random number generator was adopted from Press et al (1986) and LINPACK routines were used for matrix manipulations.
Table 1—Parameters for Trackers

\( n=5 \)

\[
\begin{align*}
X &= \begin{pmatrix} 1 & -2 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 2 & 1 \end{pmatrix} \\
Z &= \begin{pmatrix} 1 & -2 \\ 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{pmatrix} \\
\alpha &= (5, 3, 2) \\
D &= \begin{pmatrix} 0.5 & 0.0 \\ 0.0 & 0.5 \end{pmatrix}
\end{align*}
\]

The two settings for \( \sigma^2 \) are 0.5 and 2.0.

\( n=10 \)

\[
\begin{align*}
X &= \begin{pmatrix} 1 & -4 & 0 \\ 1 & -3 & 0 \\ 1 & -2 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 3 & 1 \\ 1 & 4 & 1 \\ 1 & 5 & 1 \end{pmatrix} \\
Z &= \begin{pmatrix} 1 & -4 \\ 1 & -3 \\ 1 & -2 \\ 1 & -1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 1 & 5 \end{pmatrix} \\
\alpha &= (5, 3, 2) \\
D &= \begin{pmatrix} 0.5 & 0.0 \\ 0.0 & 0.5 \end{pmatrix}
\end{align*}
\]

The two settings for \( \sigma^2 \) are 2.0 and 8.0.
Table 2 - Simulations Run with only Trackers Present

<table>
<thead>
<tr>
<th>m</th>
<th>n</th>
<th>( \sigma^2 )</th>
<th>signif</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>5</td>
<td>0.5</td>
<td>.10</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>0.5</td>
<td>.15</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2.0</td>
<td>.10</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2.0</td>
<td>.15</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2.0</td>
<td>.10</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2.0</td>
<td>.15</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>8.0</td>
<td>.10</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>8.0</td>
<td>.15</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>0.5</td>
<td>.10</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>0.5</td>
<td>.15</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>2.0</td>
<td>.10</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>2.0</td>
<td>.15</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>2.0</td>
<td>.10</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>2.0</td>
<td>.15</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>8.0</td>
<td>.10</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>8.0</td>
<td>.15</td>
</tr>
</tbody>
</table>
Figure 1 - Graph of Expected Values for Non-Trackers

\( \lambda = (5, -4, 2) \)

\( \lambda = (7, 0, 1) \)

\( n=5 \)

\( n=10 \)

--- Non-trackers

--- Trackers

1 observation 5

1 observation 10
Table 3 - Simulations Run with Non-Trackers Present

<table>
<thead>
<tr>
<th>m</th>
<th>n</th>
<th>ntr</th>
<th>$\sigma^2$</th>
<th>$\alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>A</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>B</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>1</td>
<td>2.0</td>
<td>A</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>1</td>
<td>2.0</td>
<td>B</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2</td>
<td>0.5</td>
<td>A</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2</td>
<td>0.5</td>
<td>B</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>2.0</td>
<td>A</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>2.0</td>
<td>B</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>2.0</td>
<td>A</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>2.0</td>
<td>B</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>8.0</td>
<td>A</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>8.0</td>
<td>B</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>A</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>B</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>1</td>
<td>2.0</td>
<td>A</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>1</td>
<td>2.0</td>
<td>B</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>3</td>
<td>0.5</td>
<td>A</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>3</td>
<td>0.5</td>
<td>B</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>3</td>
<td>2.0</td>
<td>A</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>3</td>
<td>2.0</td>
<td>B</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>1</td>
<td>2.0</td>
<td>A</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>1</td>
<td>2.0</td>
<td>B</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>1</td>
<td>8.0</td>
<td>A</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>1</td>
<td>8.0</td>
<td>B</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>3</td>
<td>2.0</td>
<td>A</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>3</td>
<td>2.0</td>
<td>B</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>3</td>
<td>8.0</td>
<td>A</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>3</td>
<td>8.0</td>
<td>B</td>
</tr>
</tbody>
</table>

The significance level used was 0.10. A indicates $\alpha=(5, -4, 2)$; B indicates $\alpha=(7,0,1)$. Ntr stands for the number of non-trackers present.
Figure 2 - Outline for Computer Algorithm

RANCOEF2

Read parameters

Begin a simulation
Generate y for trackers
Generate y for non-trackers
Call FITRCB2

Initial estimate of population parameters
E-M algorithm to improve above estimates using previously determined convergence criterion

Call FIND
Calculate $d_i$

Is max $d_i$ large?

YES

Throws out the non-tracker
Call FITRCB2 to recalculate parameter estimates of tracking population

Record parameter estimates for the simulation

NO

Last simulation?

YES

Calculate overall parameter estimates, sensitivity and specificity for each simulation

NO
Chapter 3 - Conclusion

The first observation, obvious by looking at Table 4, is that when no non-trackers are present the algorithm is excellent. The average specificity among repetitions stays above 99% for each type of simulation indicating that the program has no problem identifying a tracker when it really is a tracker. From Table 5 we see that the parameter estimates of $\alpha$, $\sigma^2$ and $D$ are very accurate. When signif is changed from 0.10 to 0.15 while all other parameters are held constant it is true that specificity is slightly better at the 0.10 level. More interesting is the fact that the percentage of repetitions that throw out a tracker increases as signif increases. The overall average for this percentage at the 0.10 level is 0.0775 and for 0.15 it is 0.10. This is what we would expect to happen since our p-value for $\max D_i$ is being compared to the signif level to identify non-trackers and possibly eliminate them.

When non-trackers are included the results are much more interesting. From Table 6 we see that again the average specificity among repetitions remains above 99% for each type of simulation. When we examine the sensitivity, it is easy to see that non-trackers were correctly identified with best accuracy when only one non-tracker was present and when it was significantly different from the
tracking population. It is also obvious that increasing the number of non-trackers significantly lowers the sensitivity of the program. This indicates that identifying non-trackers is very difficult when the parameter estimates are very contaminated. As we would expect, throughout the simulations, the sensitivity levels were higher for case A, \( \alpha=(5,-4,2) \), than case B, \( \alpha=(7,0,1) \), (recall that case A non-trackers are very different from the trackers where case B non-trackers are only slightly different). Signif was held constant at 0.10 for all of the simulations in which non-trackers were included. This level was used in order to reduce the number of trackers incorrectly identified as non-trackers. Changing \( \sigma^2 \) while holding everything else constant results in only a slight change in the level of sensitivity.

As shown in Table 7, the estimate of \( D, \hat{D} \), is affected significantly by the presence of more than one non-tracker. Specifically, the entries of \( \hat{D} \) are larger than \( D \). We know that when \( \hat{D} \) is large, its inverse is small and therefore by the relationship given in equation 6, \( D_1 \) is smaller than it should be. As a result, the power of the Mahalanobis distance is being reduced. This in turn reduces the power of our algorithm. Our research did not include this, but, other suggestions such as eliminating two non-trackers at a time could be studied for the applicability to this problem.

We began this project with the intentions of designing a computer algorithm for identifying non-trackers present
in a population from a balanced set of data. Although theoretically sound, some algorithms do not attain the practical application desired. For professionals, this is not discouraging but rather a way of opening other areas of study. An investigation of the influence function seems to be a logical alternative.
### Table 4 - Results with Trackers Only

<table>
<thead>
<tr>
<th>m</th>
<th>n</th>
<th>$a^2$</th>
<th>signif</th>
<th>spec</th>
<th>std dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>5</td>
<td>0.5</td>
<td>.10</td>
<td>.9976</td>
<td>.0096</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>0.5</td>
<td>.15</td>
<td>.996</td>
<td>.0121</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2.0</td>
<td>.10</td>
<td>.9968</td>
<td>.0110</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2.0</td>
<td>.15</td>
<td>.9936</td>
<td>.0169</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2.0</td>
<td>.10</td>
<td>.996</td>
<td>.0121</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2.0</td>
<td>.15</td>
<td>.996</td>
<td>.0121</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>8.0</td>
<td>.10</td>
<td>.9968</td>
<td>.0110</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>8.0</td>
<td>.15</td>
<td>.9944</td>
<td>.0162</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>0.5</td>
<td>.10</td>
<td>.9984</td>
<td>.0055</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>0.5</td>
<td>.15</td>
<td>.998</td>
<td>.0061</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>2.0</td>
<td>.10</td>
<td>.9984</td>
<td>.0055</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>2.0</td>
<td>.15</td>
<td>.998</td>
<td>.0061</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>2.0</td>
<td>.10</td>
<td>.9988</td>
<td>.0048</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>2.0</td>
<td>.15</td>
<td>.9988</td>
<td>.0048</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>8.0</td>
<td>.10</td>
<td>.9988</td>
<td>.0048</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>8.0</td>
<td>.15</td>
<td>.9988</td>
<td>.0048</td>
</tr>
</tbody>
</table>
Table 5 - Average Parameter Estimates with Trackers Only (Selected Cases)

<table>
<thead>
<tr>
<th>Case</th>
<th>m</th>
<th>n</th>
<th>$\sigma^2$</th>
<th>$\hat{\sigma}^2$</th>
<th>std dev $\hat{\sigma}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>25</td>
<td>5</td>
<td>2.0</td>
<td>1.8791</td>
<td>0.2628</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\hat{\theta} = (0.4755, 0.0244)$</td>
<td>$\hat{\alpha} = (4.9823, 0.2869)$</td>
</tr>
<tr>
<td>2</td>
<td>50</td>
<td>5</td>
<td>2.0</td>
<td>1.9458</td>
<td>0.1986</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\hat{\theta} = (0.4949, 0.0149)$</td>
<td>$\hat{\alpha} = (5.0120, 0.1964)$</td>
</tr>
<tr>
<td>3</td>
<td>25</td>
<td>10</td>
<td>8.0</td>
<td>7.9065</td>
<td>0.8343</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\hat{\theta} = (0.4964, -0.0025)$</td>
<td>$\hat{\alpha} = (4.9264, 0.4290)$</td>
</tr>
<tr>
<td>4</td>
<td>50</td>
<td>10</td>
<td>2.0</td>
<td>1.9827</td>
<td>0.1313</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\hat{\theta} = (0.4897, -0.0217)$</td>
<td>$\hat{\alpha} = (4.9841, 0.1870)$</td>
</tr>
</tbody>
</table>

True parameter values for the above are:
- $D = \begin{pmatrix} 0.5 & 0.0 \\ 0.0 & 0.5 \end{pmatrix}$
- $\alpha = \begin{pmatrix} 5 \\ 3 \\ 2 \end{pmatrix}$
### Table 6 - Results with Non-Trackers Present

<table>
<thead>
<tr>
<th>m</th>
<th>n</th>
<th>ntr</th>
<th>( \sigma^2 )</th>
<th>( \alpha )</th>
<th>spec</th>
<th>sens</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>A</td>
<td>.99583</td>
<td>1.000</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>B</td>
<td>.99667</td>
<td>0.300</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>1</td>
<td>2.0</td>
<td>A</td>
<td>.99750</td>
<td>0.960</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>1</td>
<td>2.0</td>
<td>B</td>
<td>.99667</td>
<td>0.220</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2</td>
<td>0.5</td>
<td>A</td>
<td>.99826</td>
<td>0.260</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2</td>
<td>0.5</td>
<td>B</td>
<td>.99913</td>
<td>0.100</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2</td>
<td>2.0</td>
<td>A</td>
<td>.99652</td>
<td>0.160</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>2</td>
<td>2.0</td>
<td>B</td>
<td>.99730</td>
<td>0.080</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>2.0</td>
<td>A</td>
<td>.99250</td>
<td>0.580</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>2.0</td>
<td>B</td>
<td>.99583</td>
<td>0.080</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>8.0</td>
<td>A</td>
<td>.99583</td>
<td>0.640</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>1</td>
<td>8.0</td>
<td>B</td>
<td>.99667</td>
<td>0.160</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>2.0</td>
<td>A</td>
<td>.99391</td>
<td>0.180</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>2.0</td>
<td>B</td>
<td>.99739</td>
<td>0.030</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>8.0</td>
<td>A</td>
<td>.99739</td>
<td>0.120</td>
</tr>
<tr>
<td>25</td>
<td>10</td>
<td>2</td>
<td>8.0</td>
<td>B</td>
<td>.99826</td>
<td>0.000</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>A</td>
<td>.99796</td>
<td>1.000</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>1</td>
<td>0.5</td>
<td>B</td>
<td>.99837</td>
<td>0.600</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>1</td>
<td>2.0</td>
<td>A</td>
<td>.99878</td>
<td>1.000</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>1</td>
<td>2.0</td>
<td>B</td>
<td>.99959</td>
<td>0.360</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>3</td>
<td>0.5</td>
<td>A</td>
<td>.99745</td>
<td>0.620</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>3</td>
<td>0.5</td>
<td>B</td>
<td>.99830</td>
<td>0.267</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>3</td>
<td>2.0</td>
<td>A</td>
<td>.99872</td>
<td>0.627</td>
</tr>
<tr>
<td>50</td>
<td>5</td>
<td>3</td>
<td>2.0</td>
<td>B</td>
<td>.99915</td>
<td>0.093</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>1</td>
<td>2.0</td>
<td>A</td>
<td>.99714</td>
<td>1.000</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>1</td>
<td>2.0</td>
<td>B</td>
<td>.99796</td>
<td>0.320</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>1</td>
<td>8.0</td>
<td>A</td>
<td>.99510</td>
<td>1.000</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>1</td>
<td>8.0</td>
<td>B</td>
<td>.99959</td>
<td>0.240</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>3</td>
<td>2.0</td>
<td>A</td>
<td>.99915</td>
<td>0.227</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>3</td>
<td>2.0</td>
<td>B</td>
<td>.99787</td>
<td>0.067</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>3</td>
<td>8.0</td>
<td>A</td>
<td>.99872</td>
<td>0.293</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>3</td>
<td>8.0</td>
<td>B</td>
<td>.99787</td>
<td>0.047</td>
</tr>
</tbody>
</table>

The significance level used was 0.10. A indicates \( \alpha=(5, -4, 2) \); B indicates \( \alpha=(7,0,1) \). Ntr stands for the number of non-trackers present.
Table 7 - Average Parameter Estimates with Non-Trackers Present
(Selected Cases from non-tracker \( \alpha=(5,-4,2) \))

Case 1: \( m=25 \) \( n=5 \) \( NTR=2 \) \( \sigma^2=2.0 \)
\[
\hat{\sigma}^2 = 1.8667 \quad \text{with std dev} = 0.3681
\]
\[
\hat{\theta} = \begin{pmatrix} 0.5048 & -0.0211 \\ -0.0211 & 3.5484 \end{pmatrix} \quad \alpha = \begin{pmatrix} \text{avg} \\ \text{std dev} \end{pmatrix} = \begin{pmatrix} 4.9932 \\ 0.2935 \end{pmatrix}
\]

Case 2: \( m=50 \) \( n=5 \) \( NTR=3 \) \( \sigma^2=2.0 \)
\[
\hat{\sigma}^2 = 1.8538 \quad \text{with std dev} = 0.2443
\]
\[
\hat{\theta} = \begin{pmatrix} 0.5280 & 0.0032 \\ 0.0032 & 1.4190 \end{pmatrix} \quad \alpha = \begin{pmatrix} \text{avg} \\ \text{std dev} \end{pmatrix} = \begin{pmatrix} 4.9827 \\ 0.2001 \end{pmatrix}
\]

Case 3: \( m=25 \) \( n=10 \) \( NTR=2 \) \( \sigma^2=8.0 \)
\[
\hat{\sigma}^2 = 7.8287 \quad \text{with std dev} = 0.7751
\]
\[
\hat{\theta} = \begin{pmatrix} 0.4989 & -0.0948 \\ -0.0948 & 3.6280 \end{pmatrix} \quad \alpha = \begin{pmatrix} \text{avg} \\ \text{std dev} \end{pmatrix} = \begin{pmatrix} 5.0112 \\ 0.3302 \end{pmatrix}
\]

Case 4: \( m=50 \) \( n=10 \) \( NTR=3 \) \( \sigma^2=2.0 \)
\[
\hat{\sigma}^2 = 1.9294 \quad \text{with std dev} = 0.1603
\]
\[
\hat{\theta} = \begin{pmatrix} 0.5008 & 0.0145 \\ 0.0145 & 2.6450 \end{pmatrix} \quad \alpha = \begin{pmatrix} \text{avg} \\ \text{std dev} \end{pmatrix} = \begin{pmatrix} 4.9727 \\ 0.1605 \end{pmatrix}
\]

True parameter values for the above are:

\[
\text{signif} = 0.10
\]
\[
\begin{pmatrix} \text{avg} \\ \text{std dev} \end{pmatrix} = \begin{pmatrix} 5 \\ 3 \end{pmatrix}
\]
Appendix 1

In order to find the mean of our multivariate normal distribution we set the general form of this distribution, where $b_i$ is the random variable, equal to our distribution, substitute for $A$ and solve for $\mu$ directly.

Since,

$$\frac{-\left[(b_i - \mu)^T A (b_i - \mu)\right]}{2} = -\frac{b_i^T A b_i - 2\mu^T A b_i + \mu^T A \mu}{2},$$

and we have,

$$A = \frac{\left[Z_i^T Z_i + D^{-1} \sigma^2\right]}{\sigma^2}.$$

Now, setting the "linear term" from above equal to the "linear term" of our distribution.
\[-2\mu^T A b_i = \frac{-2(y_i - x_i a)^T z_i b_i}{\sigma^2}\]

\[= \frac{\mu^T (z_i^T z_i + D^{-1} \sigma^2)}{\sigma^2} = \frac{(y_i - x_i a)^T z_i}{\sigma^2}\]

\[= \mu^T = (y_i - x_i a)^T z_i \left(z_i^T z_i + D^{-1} \sigma^2\right)^{-1}\]

\[= \mu = \left(z_i^T z_i + D^{-1} \sigma^2\right)^{-1} z_i \left(y_i - x_i a\right)\]
Appendix 2

We must show that
\[ \frac{\partial}{\partial D} \, b_i^T D^{-1} b_i = - D^{-1} b_i b_i^T D \]

We use the fact that
\[ b_i^T D^{-1} b_i = \text{tr} \, D^{-1} b_i b_i^T = \text{tr} \, D^{-1} A \]
where \( A = b_i b_i^T \). Both \( D^{-1} \) and \( A \) are symmetric. We know that
\[ \frac{\partial d_{ij}}{\partial d_{rc}} = - d_{ir} d_{cj} \]
where \( d_{rc} \) is the \( r,c \)th entry in \( D \) and \( d_{ir} \) is the \( i,r \)th entry in \( D^{-1} \). Then
\[ \frac{\partial b_i^T D^{-1} b_i}{\partial d_{rc}} = \frac{\partial}{\partial d_{rc}} \text{tr} \, D^{-1} A = \frac{\partial}{\partial d_{rc}} \left( \sum_i \sum_l d_{il} a_{li} \right) \]
\[ = - \sum_i \sum_l d_{ir} d_{cl} a_{li} = - \sum_i d_{ir} \left( \sum_l d_{cl} a_{li} \right) \]

The sum in parentheses in the last expression is the \( c,i \)th entry in \( D^{-1} A \), or the \( i,c \)th entry in \( AD^{-1} \) using the symmetry of \( A \) and \( D^{-1} \). Replacing \( d_{ir} \) by \( d_{ri} \) using symmetry, we have
\[ = - \sum_i d_{ri} \left( AD^{-1} \right)_{ic} \]

Since this last expression is just the \( r \)th row of \( D^{-1} \) times the \( c \)th column of \( AD^{-1} \), we have that
\[ \frac{\partial}{\partial D} \, b_i^T D^{-1} b_i = - D^{-1} b_i b_i^T D^{-1} \]
Appendix 3 - Programs for Computer Simulations

CALLING PROGRAM TO GENERATE DATA WITH TRACKERS AND NONTRACKERS.


AFTER THE INITIAL FIT, THE PROGRAM USES THE ROUTINE FINDEM TO FIND NONTRACKERS. AT THE END, PROGRAM PRINTS ESTIMATED PARAMETER VALUES FOR TRACKERS, AND A LIST OF ID NUMBERS FOR SUSPECTED NONTRACKERS.

******** BALANCED DATA DESIGNS ********

IMPLICIT REAL*8 (A-H,Q-Z)
INTEGER N,M,P,K,PN,KN,MTR,MNTR
DIMENSION X(20,6),Z(20,6)
DIMENSION Y(20,200),ALPH(6),B(6,200),D(6,6)
DIMENSION XN(20,6) ,ZN(20,6)
DIMENSION ALFBAR(6),DBAR(6,6),ALFDEV(6)
INTEGER IUSE(200)
COMMON /DATAS/X,Z,Y
 COMMON /PARAMS/ ALPH,B,SIGMA2,D
 COMMON /ITCON/MAXIT
 COMMON /USEME/IUSE
DIMENSION TALPH(6) ,TD(6,6),TQ(6,6),XA(20),BT(6)
DIMENSION TALPHN(6),TDN(6,6),TQN(6,6),XAN(20)
DIMENSION WORK(6),JTV(6)
ALL INPUT READ FROM FILE ON CHANNEL 3

IUSE(200) IS AN INTEGER VECTOR WHERE IUSE(I)=1 MEANS THE INDIVIDUAL I SHOULD BE USED IN FITTING
IUSE(I)=0 MEANS DO NOT USE INDIV. I IN FITTING MODEL

READ IN DIMENSIONS N=# OBSERVATIONS PER INDIVIDUAL
M = # OF INDIVIDUALS
P = DIMENSION OF ALPHA (FIXED EFFECTS)
K = DIMENSION OF B (RANDOM EFFECTS)
PN=DIM OF ALPHA FOR NONTRACKERS
KN=DIM OF B FOR NONTRACKERS
MTR = NUM OF TRACKERS

*** GET PARAMETERS FOR TRACKERS AND NONTRACKERS ***

READ (3,* ) N,M,P,K,PN,KN,MTR MTR=M-MTR
READ NUMITR = NUMBER OF REPETITIONS OF SIMULATION WRITE (9,* ) N,M,P,K,MTR',N,M,P,K,MTR READ (3,* ) NUMITR READ (3,* ) MAXIT,CONV
GET MATRIX X FOR TRACKERS, XN FOR NONTRACKERS
DO 10 I=1,N
 READ (3,* ) (X(I,J),J=1,P),(XN(I,J),J=1,PN)
 CONTINUE
GET MATRIX Z FOR TRACKERS, ZN FOR NONTRACKERS
DO 15 I=1,N
 READ (3,* ) (Z(I,J),J=1,K),(ZN(I,J),J=1,KN)
CONTINUE
C GET TRUE VALUES OF ALPHA, STORED IN TALPH
READ (3,*) (TALPH(I),I=1,P),(TALPHN(I),I=1,PN)
WRITE (9,*) ' ALPH FOR NONTR'
WRITE(9,16) (TALPHN(I),I=1,PN)
FORMAT(3(1X,F10.4))
C GET TRUE VALUE OF MEASUREMENT VARIANCE, SIGMA2, STORED AS TSIG2
READ (3,*) TSIG2,TSIG2N
TSIG=DSQRT(TSIG2)
TSIGN=SQRT(TSIG2N)
C GET TRUE VALUE OF COVARIANCE MATRIX D FOR RANDOM EFFECTS
DO 20 I=1,K
READ (3,*) (TD(I,J),J=1,K)
CONTINUE
DO 25 I=1,KN
READ(3,*) (TDN(I,J),J=1,KN)
CONTINUE
READ (3,*) SIGNIF
READ (3,*) IDUM
WRITE (9,*) ' TSIG2,SIGNIF'
WRITE(9,16) TSIG2,SIGNIF
FORMAT(2(1X,F10.4))
XX=RAN3(IDUM)
C CALL CHOLESKY DECOMPOSITION TO FACTOR TD=(TQ)*(TQ)
DO 75 I=1,K
DO 72 J=1,K
TQ(I,J)=TD(I,J)
CONTINUE
JOB=0
LDA=6
CALL DCHDC(TQ,LDA,K,WORK,JPV,JOB,INFO)
C WRITE (6,*) INFO
DO 78 I=1,K-1
DO 77 J=I+1,K
TQ(J,I)=TQ(I,J)
TQ(I,J)=0.DO
CONTINUE
77 CONTINUE
C CALL CHOLESKY DECOMPOSITION TO FACTOR TDN=(TQN)*(TQN)
DO 95 I=1,KN
DO 92 J=1,KN
TQN(I,J)=TDN(I,J)
CONTINUE
JOB=0
LDA=6
CALL DCHDC(TQN,LDA,KN,WORK,JPV,JOB,INFO)
C WRITE (6,*) INFO
DO 98 I=1,KN-1
DO 97 J=I+1,KN
TQN(J,I)=TQN(I,J)
TQN(I,J)=0.DO
CONTINUE
97 CONTINUE
C STORE MEAN VECTOR X*ALPHA FOR TRACKERS
DO 150 I=1,N
**C**

**++ MANUFACTURE Y FOR NONTRACKERS II=MTR+1,M ++**

**C**

**DO 4 I=MTR+1,M**

**IUSE(I)=1**

**C**

**++ MANUFACTURE Y FOR TRACKERS II=1,MTR ++**

**C**

**DO 145 J=1,P**

**XA(I)=XA(I)+X(I,J)*TALPH(J)**

**CONTINUE**

**150**

**C**

**STORE MEAN VECTOR XN*ALPHAN FOR NONTRACKERS**

**DO 190 I=1,N**

**XAN(I)=0.DO**

**DO 185 J=1,PN**

**XAN(I)=XAN(I)+XN(I,J)*TALPHN(J)**

**CONTINUE**

**185**

**CONTINUE**

**190**

**C**

**++++ BEGIN REPETITIONS, CREATING DATA FOR ++**

**C**

**++++ TRACKERS, NONTRACKERS; FITTING MODEL +++++**

**C**

**++++ AND FINDING NON-TRACKERS**

**C**

**SIGBAR=0.000**

**SIGDEV=0.000**

**DO 200 I=1,P**

**ALFBAR(I)=0.000**

**200**

**ALFDEV(I)=0.000**

**DO 210 I=1,K**

**DO 205 J=1,K**

**DBAR(I,J)=0.000**

**205**

**CONTINUE**

**210**

**CONTINUE**

**SUMSN =0.000**

**SUMSN2=0.000**

**SUMSP =0.000**

**SUMSP2=0.000**

**DO 3000 III=1,NUMITR**

**C**

**DO 4 I=1,M**

**IUOE(I)=1**

**C**

**++++ MANUFACTURE Y FOR TRACKERS II=1,MTR ++++**

**C**

**DO 300 II=1,MTR**

**MANUFACTURE B FOR THE II INDIVIDUAL**

**CALL MULTNO(TQ,6,K,BT,IDUM)**

**DO 270 I=1,K**

**B(I,II)=BT(I)**

**270**

**CONTINUE**

**DO 290 I=1,N**

**Y(I,II)=XA(I)+GASDEV(IDUM)*TSIG**

**DO 280 J=1,K**

**Y(I,II)=Y(I,II)+Z(I,J)*BT(J)**

**280**

**CONTINUE**

**290**

**CONTINUE**

**300**

**CONTINUE**

**C**

**++++ MANUFACTURE Y FOR NONTRACKERS II=MTR+1,M ++++**

**C**

**IF (MTR.GE.M) GO TO 410**

**DO 400 II=MTR+1,M**

**C**

**MANUFACTURE B FOR THE II INDIVIDUAL**

**CALL MULTNO(TQ,6,KN,BT,IDUM)**

**DO 370 I=1,KN**

**B(I,II)=BT(I)**

**370**

**CONTINUE**

**C**

**RAN 0112**

**RAN 0113**

**RAN 0114**

**RAN 0115**

**RAN 0116**

**RAN 0117**

**RAN 0118**

**RAN 0119**

**RAN 0120**

**RAN 0121**

**RAN 0122**

**RAN 0123**

**RAN 0124**

**RAN 01250**

**RAN 0126**

**RAN 0127**

**RAN 0128**

**RAN 0129**

**RAN 0130**

**RAN 0131**

**RAN 0132**

**RAN 0133**

**RAN 0134**

**RAN 0135**

**RAN 0136**

**RAN 0137**

**RAN 0138**

**RAN 0139**

**RAN 0140**

**RAN 0141**

**RAN 0142**

**RAN 0143**

**RAN 0144**

**RAN 0145**

**RAN 0146**

**RAN 0147**

**RAN 0148**

**RAN 01490**

**RAN 0150**

**RAN 0151**

**RAN 0152**

**RAN 0153**

**RAN 0154**

**RAN 0155**

**RAN 0156**

**RAN 0157**

**RAN 0158**

**RAN 0159**

**RAN 0160**

**RAN 0161**

**RAN 0162**

**RAN 0163**

**RAN 0164**

**RAN 01650**

**RAN 0166**

**RAN 0167**

**RAN 0168**

**RAN 0169**

**RAN 0170**

**RAN 0171**
CONTINUE
DO 390 I=1,N
   Y(I,II)=XAN(I)+GASDEV(IDUM)*TSIG
   DO 380 J=1,KN
      Y(I,II)=Y(I,II)+ZN(I,J)*BT(J)
   CONTINUE
390 CONTINUE
400 CONTINUE

CALL FITTING ROUTINE FOR BALANCED DATA
CALL FITRCB(N,M,P,K)
CALL FINDEM(N,M,P,K,SIGNIF)
WRITE(9,*)'THE FOLLOWING INDIVIDUALS WERE IDENTIFIED AS +NON-TRACKERS'
NUMTR=0
DO 510 I=1,M
   NUMTR=NUMTR+IUSE(I)
510 IF (IUSE(I) .EQ. 0) WRITE(9,*) I
   WRITE (9,*) , NUMBER OF TRACKERS ',NUMTR
   IDCTR=0
   IDCNTR=0
   DO 515 I=MTR+1,M
   IDCTR=IDCTR + IUSE(I)
   IF (MTR+1 .GT. M) THEN
      SENS=-1.
   ELSE
      SENS=DFLOAT(IDCTR)/DFLOAT(M-MTR)
      SUMSN=SUMSN + SENS
      SUMSN2=SUMSN2 + (SENS**2)
      ENDIF
   600 SPEC=DFLOAT(IDCTR)/DFLOAT(MTR)
   SUMSP=SUMSP + SPEC
   SUMSP2=SUMSP2 + (SPEC**2)
   WRITE (9,*),ESTIMATED ALPHAS'
   DO 605 I=1,P
      WRITE (9,604) ALPH(I)
   604 WRITE (9,604) ALPH(I)
   605 FORMAT (1X,F10.4)
   CONTINUE
   WRITE (9,611) SIGMA2
   611 FORMAT (' SIGMA2',F10.5)
   DO 615 I=1,K
      WRITE (9,617) (D(I,J),J=1,K)
   617 WRITE (9,617) (D(I,J),J=1,K)
   615 FORMAT (' D',6(1X,F9.4))
   CONTINUE
   THIS WILL CALC OVERALL PARAMS FOR EACH SIMULATION
   SIGBAR=SIGBAR + SIGMA2
   SIGDEV=SIGDEV + (SIGMA2**2)
   DO 620 I=1,P
      ALFBAR(I)=ALFBAR(I)+ALPH(I)
   620 ALFDEV(I)=ALFDEV(I)+ALPH(I)**2
   DO 650 I=1,K
   650 DBAR(I,J)=DBAR(I,J)+D(I,J)
   CONTINUE
CONTINUE
C
+++
+++
C
C CALCULATE STATS FOR SENSITIVITY AND SPECIFICITY
C
C RNITR=DFLOAT(NUMITR)
SPMEAN=SUMSP/RNITR
SPSIG=SQRT((RNITR*SUMSP2)-(SUMSP**2))/(RNITR*(RNITR-1.))
IF(MTR+1.GT.M) GO TO 750
SMean=SUMSN/RNITR
SNSIG=SQRT((RNITR*SUMSN2)-(SUMSN**2))/(RNITR*(RNITR-1.))
WRITE(9,736) SNMEAN,SNSIG
736 FORMAT (' SENSITIVITY MEAN ',F8.5,' STD DEV ',F8.5)
750 WRITE(9,737) SPMEAN,SPSIG
737 FORMAT (' SPECIFICITY MEAN ',F8.5,' STD DEV ',F8.5)
C
C WRITE OUT PARAMS FOR THE SIMULATION
C
WRITE(9,*)'OVERALL ESTIMATE OF SIGMA-SQUARED'
SIGDEV=SQRT((RNITR*SIGDEV)-(SIGBAR**2))/(RNITR*(RNITR-1.))
SIGBAR=SIGBAR/RNITR
WRITE(9,604) SIGBAR
WRITE(9,*)'WITH STANDARD DEVIATION'
WRITE(9,604) SIGDEV
WRITE(9,*)'OVERALL ESTIMATED ALPHA'
DO 800 I=1,P
ALFDEV(I)=SQRT((RNITR*ALFDEV(I))-(ALFBAR(I)**2))/(RNITR*(RNITR-1.))
ALFBAR(I)=ALFBAR(I)/RNITR
800 WRITE(9,604) ALFBAR(I)
DO 805 I=1,P
WRITE(9,604) ALFDEV(I)
WRITE(9,*)'OVERALL ESTIMATE OF D'
DO 820 I=1,K
DO 810 J=1,K
DBAR(I,J)=DBAR(I,J)/RNITR
810 CONTINUE
820 CONTINUE
WRITE(9,617) (DBAR(I,J) ,J=1,K)
1000 STOP
END

C SUBROUTINE TO PRODUCE MULTIVARIATE NORMAL VECTOR BT WITH
C COVARIANCE MATRIX GIVEN BY TQ*TRAN(TQ)
C DEFINED LENGTH OF VECTOR IS LDA. USED LENGTH IS K.
C
SUBROUTINE MULTNO(TQ,LDA,K,BT,IDUM)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION TQ(LDA,LDA),BT(LDA)
DIMENSION 2(20)
DO 10 I=1,K
Z(I)=GASDEV(IDUM)
10 CONTINUE
DO 20 I=1,K
34
RAN0232
RAN0233
RAN0234
RAN0235
RAN0236
RAN0237
RAN0238
RAN0239
RAN0240
RAN0241
RAN0242
RAN0243
RAN0244
RAN0245
RAN0246
RAN0247
RAN0248
RAN0249
RAN0250
RAN0251
RAN0252
RAN0253
RAN0254
RAN0255
RAN0256
RAN0257
RAN0258
RAN0259
RAN0260
RAN0261
RAN0262
RAN0263
RAN0264
RAN0265
RAN0266
RAN0267
RAN0268
RAN0269
RAN0270
RAN0271
RAN0272
RAN0273
RAN0274
RAN0275
RAN0276
RAN0277
RAN0278
RAN0279
RAN0280
RAN0281
RAN0282
RAN0283
RAN0284
RAN0285
RAN0286
RAN0287
RAN0288
RAN0289
RAN0290
RAN0291
```
BT(I)=0.DO
DO 15 J=1,I
    BT(I)=BT(I)+TQ(I,J)*Z(J)
15 CONTINUE
RETURN
END

C FUNCTION GASDEV PRODUCES A STANDARD NORMAL DEVIATE
FUNCTION GASDEV(IDUM)
IMPLICIT REAL*8 (A-H,O-Z)
DATA ISET/O/
IF (ISET.EQ.0) THEN
    V1=2.*RAN3(IDUM)-1.
    V2=2.*RAN3(IDUM)-1.
    R=V1**2+V2**2
    IF(R.GE.1.)GO TO 1
    FAC=DSQRT(-2.*DLOG(R)/R)
    GSET=V1*FAC
    GASDEV=V2*FAC
    ISET=1
ELSE
    GASDEV=GSET
    ISET=0
ENDIF
RETURN
END

FUNCTION RAN3 PRODUCES A UNIFORM (0,1) RANDOM DEVIATE
FUNCTION RAN3(IDUM)
IMPLICIT REAL*8 (A-H,O-Z)
IMPLICIT REAL*4(M)
PARAMETER (MBIG=4000000.,MSEED=1618033.,MZ=0.,FAC=2.5E-7)
PARAMETER (MBIG=1000000000,MSEED=161803398,MZ=0,FAC=1.E-9)
DIMENSION MA(55)
DATA IFF /0/
IF(IDUM.LT.0.OR.IFF.EQ.0)THEN
    IFF=1
    MJ=MSEED-IABS(IDUM)
    MJ=MOD(MJ,MBIG)
    MA(55)=MJ
    MK=1
    DO 11 I=1,54
        II=MOD(21*I,55)
        MA(II)=MK
        MK=MJ-MK
        IF(MK.LT.M2)MK=MK+MBIG
        MJ=MA(II)
    CONTINUE
    DO 12 I=1,4
        MA(I)=MA(I)-MA(1+MOD(I+30,55))
        IF(MA(I).LT.MZ)MA(I)=MA(I)+MBIG
    CONTINUE
    INEXT=0
    INEXTP=31
    IDUM=1
ENDIF
INEXT=INEXT+1
IF(INEXT.EQ.56)INEXT=1
```
INEXTP = INEXTP + 1
IF(INEXTP.EQ.56) INEXTP = 1
MJ = MA(INEXT) - MA(INEXTP)
IF(MJ.LT.MZ) MJ = MJ + MBIG
MA(INEXT) = MJ
RAN3 = MJ * FAC
RETURN
END
FITRCB is the routine for fitting the population parameters for trackers. The equations are taken from Diem and Liukkonen (1988) and their derivations appear in Chapter 1 of this paper.

The first part of this program calculates the initial estimates of the b's, alpha, sigma squared, and the d's. The e-m algorithm is then implemented with the above estimates in order to iteratively improve the estimates of the parameters.

SUBROUTINE FITRCB(N,M,P,K)
IMPLICIT REAL *8 (A-H,Q-Z)
INTEGER N,M,P,K
INTEGER IPVT(6)
DIMENSION X(20,6),Z(20,6),Y(20,200)
DIMENSION D(6,6),B(6,200),ALPH(6),DET(2)
DIMENSION XTX(6,6),ZTZ(6,6),ZTZI(6,6),ZZZ(6,20)
DIMENSION XT(6,20),XALPH(20),DIFF(6,6)
DIMENSION XTX(6,6),SUMI(6,6),SUMIZT(6,20),YDIFF(20,200)
INTEGER ITUSE(200)
COMMON/DATAS/X,Z,Y
COMMON/PARAMS/ALPH,B,SIGMA2,D
COMMON/ITCON/MAXIT,IFLAG,CONV
COMMON/USEME/IUSE
DATA IFRST/1/
COMMON /FIND/ZT,XALPH
IFLAG=9
FIRST calculate useful quantities
THE following gives trans(x)*x
DO 10 I=1,P
DO 5 J=1,P
XTX(I,J)=DDOT(N,X(1,I),1,X(1,J),1)
10 CONTINUE
5 CONTINUE
THE following gives two copies of trans(z)*z
DO 20 I=1,K
DO 15 J=1,K
ZTZ(I,J)=DDOT(N,Z(1,I),1,Z(1,J),1)
15 CONTINUE
20 CONTINUE
NEED to calculate inv(trans(z)*z)*trans(z).
FIRST need to get trans(z) then use LINPACK
FACTOR and solver. Above will be stored in zzz
DO 30 I=1,K
DO 25 J=1,N
ZZZ(I,J)=ZT(I,J)
30 CONTINUE
CALL DGEFA(ZTZI,6,K,IPVT,INFO)
IF (INFO.NE. 0) GO TO 500
DO 35 I=1,N
CALL DGESL(ZTZI,6,K,IPVT,ZZZ(1,I),0)
CONTINUE

C

NOW CALCULATE THE INVERSE
C
CALL DGEDI(ZTZI,6,K,IPVT,DET,WORK,1)

C

DM=0.DO
DO 39 J=1,M
DM=DM+DFLOAT(IUSE(J))
IDM=INT(DM)

C

SUM THE Y’S
C
DO 45 I=1,N
YSUM(I)=0.DO
IF (IUSE(J).EQ.0) GO TO 40
YSUM(I)=YSUM(I)+Y(I,J)
CONTINUE

C

INITIALIZE DSUM, DIFF AND SIGSUM TO ZERO
C
DO 55 I=1,K
DO 50 J=1,K
DSUM(I,J)=0.DO
CONTINUE

C

DO 57 I=1,N
DIFF(I)=0.DO
CONTINUE

C

SIGSUM=0.DO

C

THIS ROUTINE CALCULATES THE INITIAL EST OF ALPHA
THE SOLUTION IS STORED IN ALPH
C

DO 60 I=1,P
ALPH(I) = DDOT(N,X(1,I),1,YSUM,1)/DM
CONTINUE

C

CALL DGEFA(XTX,6,P,IPVT,INFO)
IF (INFO.NE.0) GO TO 501
CALL DGESL(XTX,6,P,IPVT,ALPH,0)

C

CALCULATE AND STORE THE INITIAL EST OF X*ALPH
C
DO 70 I=1,P
DO 65 J=1,N
XT(I,J)=X(J,I)
CONTINUE

C

DO 75 I=1,N
XALPH(I)=DDOT(P,XT(1,I),1,ALPH,1)
CONTINUE

C

THIS CALCULATES THE INITIAL B’S
C
DO 120 J=1,M
IF (IUSE(J).EQ.O) GO TO 120
   DO 90 I=1,N
   DIFF(I)=I(I,J)-XALPH(I)
90 CONTINUE
   DO 95 I=1,K
   B(I,J)=0.0D0
   DO 92 L=1,N
      B(I,J)=B(I,J)+ZZZ(I,L)*DIFF(L)
92 CONTINUE
   DO 105 I=1,K
      DO 100 L=1,K
         DSUM(I,L)=DSUM(I,L)+(B(I,J)*B(L,J))
100 CONTINUE
105 CONTINUE
   SIGSUM=SIGSUM+DDOT(N,DIFF,1,DIFF,1)
   CONTINUE
C THE INITIAL EST OF SIGMA SQUARED IS IN SIGMA2
   SIGMA2=SIGSUM/DFLOAT(IDM*N)-P-(K*IDM)+K)
C CALCULATE AND STORE THE INITIAL EST OF D
   DO 130 I=1,K
      DO 125 J=1,K
         DSUM(I,J)=0.0D0
      125 CONTINUE
      CALL DGEFA(D,6,K,IPVT,INFO)
      IF (INFO.NE.0) GO TO 502
      CALL DGEDI(D,6,K,IPVT,DET,WORK,1)
      C D NOW CONTAINS INV(D)
      SIGSUM=0.0D0
      CALL DGEFA(D,6,K,IPVT,INFO)
      IF (INFO.NE.0) GO TO 502
      CALL DGEDI(D,6,K,IPVT,DET,WORK,1)
C D NOW CONTAINS INV(D)
   C DO 150 I=1,K
      DO 145 J=1,K
         SUMI(I,J)=ZTZ(I,J)+(SIGMA*ZTZI(I,J))
145 CONTINUE
150 CONTINUE
CALL DGEFA(SUMI,6,K,IPVT,INFO)
IP(INFO.NE.0) GO TO 503
CALL DGEDI(SUMI,6,K,IPVT,DET,WORK,1)
DO 160 I=1,K
DO 155 J=1,N
SUMIZT(I,J)=DDOT(K,SUMI(1,I),1,ZT(1,J),1)
155 CONTINUE
160 CONTINUE
C
C CALCULATE THE IMPROVED EST OF THE B'S
C
DO 190 J=1,M
IF (IUSE(J).EQ.0) GO TO 190
DO 165 I=1,N
DIFF(I)=Y(I,J)-XALPH(I)
165 CONTINUE
170 CONTINUE
DO 185 I=1,K
DO 170 L=1,N
B(I,J)=O.DO
DO 170 L=1,N
B(I,J)=B(I,J)+(SUMIZT(I,L)*DIFF(L))
170 CONTINUE
180 CONTINUE
185 CONTINUE
C
C M-STEP
C
RE-CALCULATE THE ALPHAS
C
DO 195 I=1,N
195 CONTINUE
DO 205 J=1,M
YSUM(I)=O.DO
DO 200 I=1,N
YDIFF(I,J)=Y(I,J)-(DDOT(K,ZT(I,I),1,B(1,J),1))
YSUM(I)=YSUM(I)+YDIFF(I,J)
200 CONTINUE
205 CONTINUE
DO 210 I=1,P
ALPH(I)=DDOT(N,X(I,I),1,YSUM,1)/DM
210 CONTINUE
CALL DGESL(XTX,6,P,IPVT,ALPH,0)
DO 215 I=1,N
XALPH(I)=DDOT(F,X(I,I),1,ALPH,1)
215 CONTINUE
C
C RE-CALCULATE THE D'S
C
DO 225 I=1,K
DO 220 J=1,K
D(I,J)=(DSUM(I,J)/DM)+(SIGMA2*SUMI(I,J))
220 CONTINUE
225 CONTINUE
C
C RE-CALCULATE SIGMA-SQUARED
C

DO 235 J=1,M
IF (IUSE(J).EQ.0) GO TO 235
DO 230 I=1,N
YDIFF(I,J)=YDIFF(I,J)-XALPH(I)
230 CONTINUE
235 CONTINUE
DO 240 J=1,M
IF (IUSE(J).EQ.0) GO TO 240
SIGSUM=SIGSUM+DDOT(N,YDIFF(1,J),1,YDIFF(1,J),1)
240 CONTINUE
TRACE=0.0DO
DO 245 I=1,K
TRACE=TRACE+DDOT(K,ZTZ(1,I),1,SUMI(1,I),1)
245 CONTINUE
SIGMA2=((SIGSUM)/DFLOAT(M*N))+(SIGMA2*TRACE)/DFLOAT(N)
U=DABS(SIGMA2-SIGOLD)
DO 255 I=1,K
DO 250 J=1,I
R=DABS(DOLD(I,J)-D(I,J))
250 CONTINUE
255 CONTINUE
IF (R.GT.U) U=R
IF (U.GT.CONV) GO TO 1100
IFLAG=1
WRITE(20,*) 'CONVERGED IN',ITER,'ITERATIONS'
1100 IF(ITER.LT.MAXIT) THEN
   GO TO 1000
ELSE
   WRITE(20,*) 'FAILED TO CONVERGE IN',MAXIT,'ITERATIONS'
ENDIF
RETURN
500 WRITE(20,*) 'THE MATRIX TRANS(Z)*Z IS NOT INVERTIBLE'
STOP
501 WRITE(20,*) 'THE MATRIX TRANS(X)*X IS NOT INVERTIBLE'
STOP
502 WRITE(20,*) ' THE MATRIX D IS NOT INVERTIBLE'
STOP
503 WRITE(20,*) ' THE MATRIX SUMI IS NOT INVERTIBLE'
STOP
END
ONCE THE POPULATION PARAMETERS HAVE BEEN
CALCULATED BY FITRCB USING ALL OBSERVATIONS
FINDEM CALCULATES THE MAXIMUM MAHALNOBIS
DISTANCE. THIS NUMBER IS COMPARED TO A
P-VALUE CALCULATED BY THE FUNCTION PVCHI
AND EITHER ELIMINATED OR KEPT. EACH TIME
AN OBSERVATION IS ELIMINATED FITRCB IS
CALLED TO RE-CALCULATE THE PARAMETERS FOR
THE 'TRACKING' POPULATION.

SUBROUTINE FINDEM (N,M,P,K,SIGNIF)
IMPLICIT REAL *8 (A-H,Q-Z)
INTEGER N,M,P,K
INTEGER JPVT(20),IPVT(20)
DIMENSION X(20,6),Z(20,6),Y(20,200)
DIMENSION D(6,6),B(6,200),ALPH(6),DET(2)
DIMENSION ZT(6,20)
DIMENSION XALPH(20),WORK(20)
DIMENSION ZDT(6,20),COV(20,20),U(20),DMH(200)
INTEGER IUSE(200)
DIMENSION YDIFF(20)
COMMON/DATAS/X,Z,Y
COMMON/PARAMS/ALPH,B,SIGMA2,D
COMMON/ITCON/MAXIT,IFLAG,CONV
COMMON/USEME/IUSE
COMMON/FIND/ZT,XALPH

FIRST NEED TO CALCULATE THE COVARIANCE MATRIX
THIS CALCULATES INV(Z*D*TRANS(Z)+SIGMA2*I)

1000 DO 10 I=1,N
   DO 5 J=1,K
      ZDT(J,I)=DDOT(K,ZT(1,I),1,D(1,J),1)
   CONTINUE
10 CONTINUE
   DO 20 I=1,N
      DO 15 J=1,N
         COV(I,J)=DDOT(K,ZDT(1,J),1,ZT(1,I),1)
      CONTINUE
20 CONTINUE
   CALL DGEFA(COV,20,N,IPVT,INFO)
   IF (INFO.NE.0) THEN
      WRITE(9,*) 'COV IS NOT INVERTIBLE'
   ENDIF
   CALL DGEDI(COV,20,N,IPVT,DET,WORK,1)

USE CHOLESKY DECOMP TO CALC MAHALANOBIS DIST
DM=0.
DMAX=-1.0
DO 30 I=1,N
   IF (IUSE(I).EQ.0) GO TO 30
   DM=DM+1.
30 CONTINUE
   DO 22 J=1,N

FIN0001 FIN0002 FIN0003 FIN0004 FIN0005 FIN0006 FIN0007 FIN0008 FIN0009 FIN0010 FIN0011 FIN0012 FIN0013 FIN0014 FIN0015 FIN0016 FIN0017 FIN0018 FIN0019 FIN0020 FIN0021 FIN0022 FIN0023 FIN0024 FIN0025 FIN0026 FIN0027 FIN0028 FIN0029 FIN0030 FIN0031 FIN0032 FIN0033 FIN0034 FIN0035 FIN0036 FIN0037 FIN0038 FIN0039 FIN0040 FIN0041 FIN0042 FIN0043 FIN0044 FIN0045 FIN0046 FIN0047 FIN0048 FIN0049 FIN0050 FIN0051 FIN0052 FIN0053 FIN0054 FIN0055 FIN0056 FIN0057
22 \ YDIFF(J) = Y(J,1) - XALPH(J) \\
25 \ DO 25 J = 1, N \\
26 \ U(J) = DDOT(N, COV(1, J), 1, YDIFF(1), 1) \\
27 \ CONTINUE \\
28 \ DMH(I) = DDOT(N, U(I), 1, YDIFF(1), 1) \\
29 \ IF(DMH(I) GT DMAX) THEN \\
30 \ \ \ \ DMAX = DMH(I) \\
31 \ \ \ \ INDEX = I \\
32 \ \ \ \ ENDIF \\
33 \ CONTINUE \\
34 \ \ PV = PVCHI(DMAX, N, DM) \\
35 \ \ IF(PV .LT. SIGNIF) THEN \\
36 \ \ \ \ IUSE(INDEX) = 0 \\
37 \ \ \ \ CALL FITRCB(N, M, P, K) \\
38 \ \ \ \ GO TO 1000 \\
39 \ \ ELSE \\
40 \ \ \ \ RETURN \\
41 \ \ \ \ ENDIF \\
42 \ \ END \\

FUNCTION PVCHI(DMAX, N, DM) 
WRITTEN BY D MOHR 10/1/89 
RETURNS PROB. MAX OF M INDEP CHI-SQUARED VARIATES 
(EACH WITH N D.F.) WILL BE GREATER THAN D. 
IMPLICIT REAL*8 (A-H,O-Z) 
INTEGER N 
DATA DL7/-.356675/ 
RN2=DFLOAT(N)/2. 
DM2=DMAX/2. 
PV=GAMMP(RN2, DM2) 
PV=DLOG(PV) 
DL7M=DL7/DM 
IF(DL7M.GT.PV) THEN 
PVCHI=.3 
ELSE 
PV=DEXP(PV*DM) 
PVCHI=1.-PV 
ENDIF 
RETURN 
END 

FUNCTION GAMMQ(A, X) 
FROM 'NUMERICAL RECIPES' 
IMPLICIT REAL*8 (A-H,O-Z) 
IF(X.LT.0. OR.A.LE.0.) PAUSE 
IF(X.LT.A+1.) THEN 
\ \ \ CALL GSER(GAMSER, A, X, GLN) 
\ \ \ GAMMQ = 1. - GAMSER 
ELSE 
\ \ \ CALL GCF(GAMMCF, A, X, GLN) 
\ \ \ GAMMQ = GAMMCF 
ENDIF 
RETURN 
END 

SUBROUTINE GSER(GAMSER, A, X, GLN) 
FROM 'NUMERICAL RECIPES' 
PARAMETER (ITMAX=100, EPS=3.E-7)
IMPLICIT REAL*8 (A-H,O-Z)

GLN=GAMMLN(A)

IF(X.LE.0.)THEN
  IF(X.LT.0.)PAUSE
  GAMSER=0.
  RETURN
ENDIF

AP=A
SUM=1./A
DEL=SUM
DO 11 N=1,ITMAX
  AP=AP+1.
  DEL=DEL*X/AP
  SUM=SUM+DEL
  IF(ABS(DEL).LT.ABS(SUM)*EPS) GO TO 1
1 CONTINUE
PAUSE 'A TOO LARGE, ITMAX TOO SMALL'
1 GAMSER=SUM*EXP(-X+A*LOG(X)-GLN)
RETURN
END

C
SUBROUTINE GCF(GAMMCF,A,X,GLN)
C FROM 'NUMERICAL RECIPES'
PARAMETER (ITMAX=100,EPS=3.E-7)
IMPLICIT REAL*8 (A-H,O-Z)
GLN=GAMMLN(A)
GOLD=0.
AO=1.
A1=X
B0=0.
B1=1.
FAC=1.
DO 11 N=1,ITMAX
  AN=FLOAT(N)
  ANA=AN-A
  AO=(A1+AO*ANA)*FAC
  BO=(B1+B0*ANA)*FAC
  ANF=AN*FAC
  A1=X*AO+ANF*A1
  B1=X*BO+ANF*B1
  IF(A1.NE.0.)THEN
    FAC=1./A1
    G=B1*FAC
    IF(ABS((G-GOLD)/G).LT.EPS) GO TO 1
    GOLD=G
  ENDIF
1 CONTINUE
PAUSE 'A TOO LARGE, ITMAX TOO SMALL'
1 GAMMCF=EXP(-X+A*LOG(X)-GLN)*G
RETURN
END

C
FUNCTION GAMMLN(XX)
C FROM 'NUMERICAL RECIPES'
IMPLICIT REAL*8 (A-H,O-Z)
REAL*8 COF(6),STP,HALF,ONE,PPF,X,TMP,SER
DATA 1OF,STP/76.18009173D0,-86.50532033D0,24.01409822D0,
* -1.231739516D0,-120858063D-2,-536382D-5,2.50662827465D0/
DATA HALF,ONE,PPF/0.5D0,1.0D0,5.5D0/
X=XX-ONE
TMP=X+FPF
TMP=(X+HALF)*LOG(TMP)-TMP
SER=ONE
DO 11 J=1,6
  X=X+ONE
  SER=SER+COF(J)/X
  CONTINUE
GAMMLN=TMP+LOG(STP*SER)
RETURN
END

FUNCTION GAMMP(A,X)
FROM 'NUMERICAL RECIPES'
IMPLICIT REAL*8 (A-H,O-Z)
IF (X.LT.0. OR. A.LE.0.) PAUSE
IF (X.LT.A+1.) THEN
  CALL GSER(GAMSER,A,X,GLN)
  GAMMP=GAMSER
ELSE
  CALL GCF(GAMMCF,A,X,GLN)
  GAMMP=1.-GAMMCF
ENDIF
RETURN
END
REFERENCES


VITA

Tamarah Crouse Dishman

Educational

University of Maryland, Baltimore County
Loyola College in Baltimore - B.S. in Mathematics 1985
University of West Florida (UWF) - Graduate Studies
University of North Florida (UNF) - Graduate Studies

Professional

UWF - Graduate Teaching Assistantship
UNF - Graduate Teaching Assistantship

Honors

National Honor Society - Andover High School
Who's Who among American High School Graduates
Pi Mu Epsilon
Who's Who Among Students in American Universities and Colleges