A COMPARISON OF ESTIMATION TECHNIQUES FOR
THE THREE PARAMETER PARETO DISTRIBUTION

THESIS

Dennis J. Charek
Major, USAF

AFIT/GSO/MA/850-3

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio
A COMPARISON OF ESTIMATION TECHNIQUES FOR
THE THREE PARAMETER PARETO DISTRIBUTION

THESIS

Dennis J. Charek
Major, USAF

AFIT/650/MA/BSD-3

Approved for public release; distribution unlimited
A COMPARISON OF ESTIMATION TECHNIQUES FOR
THE THREE PARAMETER PARETO DISTRIBUTION

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Space Operations

Dennis J. Charek, B.S., M.S.
Major, USAF

December 1985

Approved for public release; distribution unlimited
Preface

The purpose of this research is to compare the minimum distance estimation technique with the best linear unbiased estimation technique to determine which estimator provides more accurate estimates of the underlying location and scale parameter values for a given Pareto distribution. Two forms of the Kolmogorov, Anderson-Darling, and Cramer-von Mises minimum distance estimators are tested. A Monte Carlo methodology is used to generate the Pareto random variates and the resulting estimates. A mean square error comparison is then performed to evaluate which estimator provides the best results. Additionally, various sample sizes and shape parameters are also used to determine whether they have an influence on a given estimator's performance.

I wish to express my sincere appreciation to my advisor, Dr. Albert H. Moore, for his guidance and direction throughout this thesis project. I also wish to thank my reader, Lt Col Joseph Coleman, for his advice and comments which greatly supplemented my efforts during this study. In addition, I would like to thank my classmate, Capt James Porter, for his assistance and contributions to this study.

Finally, I am very grateful to my wife, Monica, for her love, tolerance, and support throughout this thesis effort. I also wish to thank my daughter, Emily, and my son, Brian, for their understanding when playtime was interrupted by homework.

Dennis J. Charek
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>ii</td>
</tr>
<tr>
<td>List of Figures</td>
<td>v</td>
</tr>
<tr>
<td>List of Tables</td>
<td>vi</td>
</tr>
<tr>
<td>Abstract</td>
<td>vii</td>
</tr>
<tr>
<td>I. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>Specific Problem</td>
<td>4</td>
</tr>
<tr>
<td>Research Question</td>
<td>5</td>
</tr>
<tr>
<td>General Approach</td>
<td>5</td>
</tr>
<tr>
<td>Sequence of Presentation</td>
<td>5</td>
</tr>
<tr>
<td>II. Estimation Techniques</td>
<td>7</td>
</tr>
<tr>
<td>Estimation</td>
<td>7</td>
</tr>
<tr>
<td>Estimator Properties</td>
<td>9</td>
</tr>
<tr>
<td>Unbiased Estimators</td>
<td>9</td>
</tr>
<tr>
<td>Consistent Estimators</td>
<td>11</td>
</tr>
<tr>
<td>Efficient Estimators</td>
<td>12</td>
</tr>
<tr>
<td>Invariant Estimators</td>
<td>12</td>
</tr>
<tr>
<td>Summary</td>
<td>13</td>
</tr>
<tr>
<td>Empirical Distribution Function (EDF)</td>
<td>14</td>
</tr>
<tr>
<td>Best Linear Unbiased Estimator (BLUE)</td>
<td>15</td>
</tr>
<tr>
<td>Minimum Distance (MD) Estimator</td>
<td>16</td>
</tr>
<tr>
<td>Kolmogorov Distance</td>
<td>19</td>
</tr>
<tr>
<td>Cramer-von Mises Distance</td>
<td>20</td>
</tr>
<tr>
<td>Anderson-Darling Distance</td>
<td>20</td>
</tr>
<tr>
<td>III. Pareto Distribution</td>
<td>22</td>
</tr>
<tr>
<td>History</td>
<td>22</td>
</tr>
<tr>
<td>Applications</td>
<td>24</td>
</tr>
<tr>
<td>Socio-economic Related Applications</td>
<td>24</td>
</tr>
<tr>
<td>Militarily Related Applications</td>
<td>25</td>
</tr>
<tr>
<td>Pareto Function</td>
<td>26</td>
</tr>
<tr>
<td>Grouping Pareto Distributions By Kind</td>
<td>27</td>
</tr>
<tr>
<td>Grouping Pareto Distributions By Parameter Number</td>
<td>28</td>
</tr>
<tr>
<td>Parameter Estimation</td>
<td>30</td>
</tr>
<tr>
<td>Best Linear Unbiased Estimator</td>
<td>31</td>
</tr>
<tr>
<td>BLUEs for Shape Greater Than Z</td>
<td>31</td>
</tr>
<tr>
<td>BLUEs for Shape Equal to or Less Than Z</td>
<td>36</td>
</tr>
<tr>
<td>Minimum Distance Estimator</td>
<td>38</td>
</tr>
<tr>
<td>IV. Monte Carlo Analysis</td>
<td>40</td>
</tr>
<tr>
<td>--------------------------</td>
<td>----</td>
</tr>
<tr>
<td>Monte Carlo Method</td>
<td>40</td>
</tr>
<tr>
<td>Monte Carlo Steps and Procedures</td>
<td>41</td>
</tr>
<tr>
<td>Step 1: Data Generation</td>
<td>41</td>
</tr>
<tr>
<td>Step 2: Estimate Computation</td>
<td>43</td>
</tr>
<tr>
<td>Step 3: Estimate Comparison</td>
<td>45</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>V. Results, Analysis and Conclusions</th>
<th>49</th>
</tr>
</thead>
<tbody>
<tr>
<td>Results</td>
<td>49</td>
</tr>
<tr>
<td>Analysis</td>
<td>52</td>
</tr>
<tr>
<td>Conclusions</td>
<td>55</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VI. Summary and Recommendations</th>
<th>56</th>
</tr>
</thead>
<tbody>
<tr>
<td>Summary</td>
<td>56</td>
</tr>
<tr>
<td>Recommendations</td>
<td>58</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Appendix A: Tables of Mean Square Errors</th>
<th>59</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appendix B: Computer Program for Estimator Comparison</td>
<td>70</td>
</tr>
<tr>
<td>Bibliography</td>
<td>105</td>
</tr>
<tr>
<td>Vita</td>
<td>109</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Pseudocode for Program BLUMO</td>
<td>48</td>
</tr>
<tr>
<td>2. Sample Table of Mean Square Errors</td>
<td>50</td>
</tr>
</tbody>
</table>
### List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. Mean Square Error for $c = 1$ and $n = 6$</td>
<td>60</td>
</tr>
<tr>
<td>II. Mean Square Error for $c = 1$ and $n = 9$</td>
<td>60</td>
</tr>
<tr>
<td>III. Mean Square Error for $c = 1$ and $n = 12$</td>
<td>61</td>
</tr>
<tr>
<td>IV. Mean Square Error for $c = 1$ and $n = 15$</td>
<td>61</td>
</tr>
<tr>
<td>V. Mean Square Error for $c = 1$ and $n = 18$</td>
<td>62</td>
</tr>
<tr>
<td>VI. Mean Square Error for $c = 2$ and $n = 6$</td>
<td>62</td>
</tr>
<tr>
<td>VII. Mean Square Error for $c = 2$ and $n = 9$</td>
<td>63</td>
</tr>
<tr>
<td>VIII. Mean Square Error for $c = 2$ and $n = 12$</td>
<td>63</td>
</tr>
<tr>
<td>IX. Mean Square Error for $c = 2$ and $n = 15$</td>
<td>64</td>
</tr>
<tr>
<td>X. Mean Square Error for $c = 2$ and $n = 18$</td>
<td>64</td>
</tr>
<tr>
<td>XI. Mean Square Error for $c = 3$ and $n = 6$</td>
<td>65</td>
</tr>
<tr>
<td>XII. Mean Square Error for $c = 3$ and $n = 9$</td>
<td>65</td>
</tr>
<tr>
<td>XIII. Mean Square Error for $c = 3$ and $n = 12$</td>
<td>66</td>
</tr>
<tr>
<td>XIV. Mean Square Error for $c = 3$ and $n = 15$</td>
<td>66</td>
</tr>
<tr>
<td>XV. Mean Square Error for $c = 3$ and $n = 18$</td>
<td>67</td>
</tr>
<tr>
<td>XVI. Mean Square Error for $c = 4$ and $n = 6$</td>
<td>67</td>
</tr>
<tr>
<td>XVII. Mean Square Error for $c = 4$ and $n = 9$</td>
<td>68</td>
</tr>
<tr>
<td>XVIII. Mean Square Error for $c = 4$ and $n = 12$</td>
<td>68</td>
</tr>
<tr>
<td>XIX. Mean Square Error for $c = 4$ and $n = 15$</td>
<td>69</td>
</tr>
<tr>
<td>XX. Mean Square Error for $c = 4$ and $n = 18$</td>
<td>69</td>
</tr>
</tbody>
</table>
Abstract

This investigation compared the minimum distance estimation technique with the best linear unbiased estimation technique to determine which technique provided more accurate estimates of the location and scale parameter values when applied to the three parameter Pareto distribution. Six distinct minimum distance estimators were developed. Of these six, two were based on the Kolmogorov distance, two were based on the Anderson-Darling distance, and two were based on the Cramer-von Mises distance. For a given sample size and Pareto shape parameter, the location and scale parameters were estimated. Additionally, varying combinations of sample sizes (6, 9, 12, 15, or 18) and shape parameters (1.0, 2.0, 3.0, or 4.0) were tested to investigate the affect of such changes.

A Monte Carlo methodology was used to generate the 1000 sample sets of Pareto random variates for each sample size - shape parameter combination with location and scale parameters both set to a value of 1. The best linear unbiased estimator and the six minimum distance estimators then provided parameter estimates based on the sample sets. Finally, these estimates were compared using the mean square error as the evaluation tool. The results of this investigation indicate that the best linear unbiased estimation technique provided more accurate estimates of location and scale for the three parameter Pareto distribution than did the minimum distance estimation techniques.
A COMPARISON OF ESTIMATION TECHNIQUES FOR
THE THREE PARAMETER PARETO DISTRIBUTION

I. Introduction

Parameter estimation is an important underlying technique in statistical analysis. Although the statistician can perform some analysis intuitively, estimation requires a specific method. For example, if a statistician is asked to analyze some sample data, he could order it in ascending order and draw a histogram reflecting the occurrence frequency of values within certain intervals. Further, from the histogram's shape, he could guess the underlying population distribution. However, he could not easily determine the parameters (e.g. mean, standard deviation) of the population. At this point, the statistician needs a method to estimate the true population parameters from the sample data. The method is called the estimator, and the approximations based on the sample are the statistics (i.e. the estimates). Mendenhall defines an estimator as "a rule which specifically states how one may calculate the estimate based upon information contained in a sample" (23:13). Using these rules, the statistician can estimate the parameters of a population distribution based on sample data drawn from the population. These estimates then summarize the properties of the population for the investigator.

One estimation technique, called the best linear unbiased estimator (BLUE), relies on a linear combination of order statistics (10:265). Order statistics are a set of variables arranged according
to their magnitudes. For instance, ordering a set of observed random variables (e.g. fastest times in an automobile race) from smallest to largest results in a set of order statistics (24:229). The best linear unbiased estimator (T) can be used to estimate an unknown population parameter (θ) where T is only dependent on the values of n independent random variables. In addition, the estimator, T, must be linear in the set of n random variables. The estimator must also display the minimum variance among linear estimators and must be unbiased (10:265-266). In simple terms, unbiased means that on the average, the value of the estimator equals the parameter being estimated (33:197). Therefore, by combining a set of order statistics in a linear fashion, one can produce estimators for the underlying population parameters. If these estimators also possess the properties of minimum variance and unbiasedness, then they are called best linear unbiased estimators.

Another parameter estimation technique is minimum distance estimation, introduced by Wolfowitz in the 1950s as a method which "in a wide variety of cases, will furnish super consistent estimators even when classical methods...fail to give consistent estimators" (33:9). A minimum distance estimator is consistent if, as the sample size increases, the probability that the estimate approaches the true value of the parameter also increases (33:199). The minimum distance estimation technique is closely related in theory to the statistical procedure called goodness of fit because a distance measure is the evaluation criteria for both procedures. In goodness of fit, one tests the sample data to identify its underlying unknown distribution. A
goodness of fit test is "a test designed to compare the sample obtained with the type of sample one would expect from the hypothesized distribution to see if the hypothesized distribution function 'fits' the data in the sample" (8:189). Certain goodness of fit tests are based on a distance measure between the sample and a hypothesized distribution with known population parameters. Minimum distance estimation, however, reverses the goodness of fit approach by assuming a probability distribution type and then finding the values that minimize the distance measure. These values become the estimates of the population parameters (18:34).

Even though the minimum distance estimation technique was developed in 1953, researchers have not extensively studied the technique until recently. Parr and Schucany reported in 1979 that the method yields "strongly consistent estimators with excellent robustness properties" (27:5) when used to estimate the location parameter of symmetric distributions (27:5). Robustness of an estimator is its ability to serve as a good estimator even when the distribution assumptions are not strictly followed (27:3). Additionally, several Air Force Institute of Technology (AFIT) students, under the guidance of Dr. Albert H. Moore, have completed thesis research projects by applying the minimum distance estimation technique to specific distributions and comparing this technique with other estimation methods. These former students include Maj McNeese, working with the generalized exponential power distribution; Capt Daniels, working with the generalized t distribution; Capt Miller, working with the three parameter Weibull distribution; Capt James, working with the three parameter gamma distribution; 2Lt
Bertrand, working with the four parameter beta distribution; and 2Lt Keffer, working with the three parameter lognormal distribution. Results have generally shown that minimum distance estimators provide better estimates (i.e., estimates closer to the actual population parameters) than the other techniques used (4:9).

The literature search reveals that the capabilities of the minimum distance estimation technique have not been compared with those of the best linear unbiased estimator with regard to the Pareto distribution, a distribution of considerable value. The Pareto distribution has a variety of uses in the commercial sector. Johnson and Kotz identify several Pareto distribution analysis areas, including city population distribution, stock price fluctuation, and oil field location (16:242). In addition to commercial users, the Air Force also uses the Pareto distribution in a number of analysis areas: time to failure of equipment components (9), maintenance service times (14), nuclear fallout particles' distribution (11), and error clusters in communications circuits (3). In sum, the Pareto distribution proves to be a distribution worthy of further investigation. Use of the minimum distance estimation technique applied to the Pareto distribution offers the researcher a chance to expand the frontier of knowledge in this area.

SPECIFIC PROBLEM

Researchers have not explored the potential of the minimum distance estimation technique to improve upon the best linear unbiased estimation technique as applied to the Pareto distribution. A comparison of the techniques in a controlled environment is needed to evaluate which
technique performs better under given circumstances. The controlled environment should specify the sample size and the value of the parameters of the underlying Pareto distribution function for each comparison attempt.

RESEARCH QUESTION

For specified parameter values and sample sizes, which estimation technique, minimum distance or best linear unbiased, performs better when applied to the Pareto distribution?

GENERAL APPROACH

Monte Carlo analysis is the analytical method to be used to make the estimation technique comparison. Monte Carlo analysis of estimation methods consists of three steps. First, one generates random variates from a specified Pareto distribution (i.e., a Pareto distribution with known parameters). Second, the two estimation techniques are used to obtain parameter estimates based on the random sample data from the first step. Third, the resulting estimates are compared to determine which estimation technique provided the better parameter estimates (4:27). The mean square error technique can be used to perform this evaluation (4:31).

SEQUENCE OF PRESENTATION

This report will proceed with five additional chapters. The second chapter will discuss the estimation techniques used in this study while the third chapter will present the Pareto Distribution. The fourth chapter will describe the Monte Carlo analysis methodology used to make
the estimation technique comparisons. The fifth chapter will present the results and conclusions of the study while the sixth chapter will provide a short summary and some recommendations for future study in this area.
II. Estimation Techniques

This chapter will first provide a discussion on estimation in general, some desirable properties of estimators, and the empirical distribution as an estimator of the true distribution. Following this discussion, the two estimation techniques to be compared in this thesis will be presented. First the best linear unbiased technique will be discussed along with its inherent properties. Then the minimum distance technique will be presented in the three distance measure forms to be used throughout the rest of this study.

Estimation is part of a larger area of study called statistical inference. The statistician makes inferences about the state of nature, or the "way things really are" (22:187), based on data gathered from experiments done to discover something about the state of nature (22:187). Lindgren then narrows his discussion of statistical problems to decision problems, eliminating the areas of experimental design and representative data gathering.

Some statistical problems, notably in business and industry, are decision problems, in which the partial information about the state of nature provided by data from experimentation is used as the basis of making an immediate decision (22:188). Lindgren then describes the general decision problem as consisting of a set or 'space' A of possible actions that might be taken, the individual 'points' of this space being the individual actions (22:188). He finally defines estimation problems as "those in which the action space A is identical with the space of parameter values that index the family
of possible states of nature" (22:188). In this case, states of nature could be described by the distribution function family members, each member being defined through its own set of parameter values.

Pritsker describes the concept of parameter estimation by presenting two supporting definitions. He first defines the 'population' as the set of data points consisting "of all possible observations of a random variable" (31:46). He then defines a 'sample' as being "only part of these observations" (31:46). A method to summarize a set of data is "to view the data as a sample which is then used to estimate the parameters of the parent or underlying population" (31:46). Runyon and Haber simply define a parameter as "a summary numerical value calculated from a population" (33:4).

Liebelt indicates that the estimation problem, defined earlier by Lindgren, is difficult to solve. In fact, because there can be many estimates regarding a problem, the solution is not unique. Therefore, the statistician begins searching for the 'best' estimate; but, since the criteria for a 'best' estimate is arbitrary, there cannot be an optimal estimate to solve all problems (21:135-136). "Each problem may require a different set of optimal criteria; the choice is always left to the user of estimation theory" (21:136). So, the search always continues for a better estimator. This thesis is a continuation of that search.

Before we continue by listing and defining some of the agreed upon properties of a good estimator, we must clarify the difference between an estimator and an estimate. Mendenhall explains that an estimator is "a rule which specifically states how one may calculate the estimate based upon information contained in a sample" (23:13).
the estimator is used to produce a particular value based on specified sample data, "the resulting predicted value is called an estimate" (23:13). Wine draws an analogy to describe the difference. He indicates the distinction between the two is the same as the difference between a function, \( f(x) \), and the evaluated functional value, \( f(c) \). "\( f(x) \) is a variable defined in some domain of \( x \), and \( f(c) \) is a constant corresponding to a specified value of \( x \) equal to constant \( c \)" (37:170-171). Before a sample is drawn, we have an estimator. After the sample is drawn, the estimator produces a particular value which is an estimate (37:171).

**ESTIMATOR PROPERTIES**

The search for better estimators continues; but, what is the criteria for determining a good estimator? Certain properties of estimators have been defined and seem to be reasonable guides for choosing good estimators, although these criteria cannot be fully "justified except on the basis of intuition" (21:136). This section will discuss four of these desirable properties. If an estimator is to be used in repeated samplings from the same population, then unbiasedness is a desirable property; otherwise, a biased estimator could possibly be found which provides better parameter estimates. Additionally, a good estimator should be consistent, efficient and invariant. Each of these properties will now be described in more detail.

**Unbiased Estimators.** The first property a good estimator to be used in repeated samplings from the same population is unbiasedness. Freeman defines an unbiased estimator as follows:
We have a population described by the density function $f(x; \theta)$, where $f$ is known and the value of the parameter is unknown. A random sample $x_1, x_2, ..., x_n$ is drawn from this population. The statistic $t(x_1, x_2, ..., x_n)$ is an unbiased estimator of the parameter $\theta$ if

$$E(t) = \theta \quad (2.1)$$

for all $n$ and for any possible value of $\theta$ (10:229).

Wine points out that this definition "requires that the mean of the sampling distribution of any statistic equals the parameter which the statistic is supposed to estimate" (37:172). In other words, the expected value of the statistic $t$ equals the parameter being estimated, where "the expected value of a random variable $x$ with density function $f(v)$ is defined as

$$E(x) = \int_{-\infty}^{\infty} v f(v) dv \quad (2.2)$$

(21:85). Freeman defines the term density function as "a function $f(x_i)$ which is connected to probability statements on the random variable $x$ by

$$p(x = x_i) = f(x_i) \quad (2.3)$$

(10:18). Looking at unbiasedness from a slightly different perspective, Liebelt says that unbiasedness "is desirable, for it states that in the absence of measurement error, and uncertainty in the estimation procedure, the estimate becomes the true value" (21:137). Freeman adds a final note concerning unbiased estimators. He indicates that for an estimator to be truly unbiased, Eq (2.1) "is required to hold for all sample sizes $n$" (10:229). There are cases when Eq (2.1) roughly holds.
only for very large sample sizes. In these cases, the estimator is merely 'asymptotically unbiased' (10:229).

Unbiasedness is an important property for an estimator to have in repeated samplings from the same population. The reason for this statement becomes apparent when one looks at what can happen if an estimator is biased. "Any estimating process used repeatedly and which on the average (mean) is not equal to the parameter leads to a sure cumulation of error in one direction" (10:229). To avoid this accumulation of error in one direction, the statistician seeks to find and use unbiased estimators. However, in a single estimation situation, unbiasedness may not be desireable. Instead, one could seek to minimize the mean square error of the estimate which could then result in a better estimate.

**Consistent Estimators.** The second property of a good estimator is that of consistency. As the sample size increases, one would want the risk associated with the estimator to decrease. "That is, the estimator ought to be better when it is based on twenty observations than when it is based on two observations" (25:172). This supposition portrays the idea of consistency. "An estimator is consistent if for a large sample there is a high probability that the estimator will be near the parameter it is intended to estimate" (5:140).

A similar definition expressed by Wine uses the idea of convergence to define a consistent estimator. An estimator, $t$, of the parameter $\theta$ is consistent if, for any small numbers $\delta$ and $\varepsilon$, "there exists an integer $n'$ such that the probability that $|t - \theta| < \varepsilon$ is greater than $1 - \delta$ for all $n > n'$ " (37:171). This definition introduces the idea of convergence by saying, "given any
small [\hat{\theta}], we can find a sample size large enough so that, for all larger sample sizes, the probability that [t] differs from the true value \theta [by] more than \hat{\theta} is as small as we please" (37:171). Therefore, the estimator, t, converges in probability to \theta (37:171). Consistency, then, implies that as sample sizes increase, the probability also increases that the estimator provides estimates which more closely approximate the true value of the parameter being estimated.

**Efficient Estimators.** The third desirable property of a good estimator is that of efficiency. Efficiency is generally used as a measure to compare two estimators. The efficiency is the ratio of their mean square errors. Mendenhall and Scheaffer indicate that the mean square error can be written as the summation of the variance and the square of the bias of an estimator (24:267).

Since variance is a measure of the dispersion of the distribution of an estimator about the parameter value, the statistician seeks an estimator with small variance. By selecting an estimator with the smaller variance, he ensures that his estimates will converge more rapidly to the true parameter value (32:155). Therefore, "one estimator is said to be more efficient than another when the variability of its sampling distribution is less" (33:198).

**Invariant Estimators.** The final property of a good estimator is that of invariance. Invariance is particularly desirable when functional transformations must be made regarding the parameter. As Freeman states:
We call a method of estimation invariant under transformation of a parameter if, when the method leads to $t$ as the estimator of $\theta$, the method also leads to $g(t)$ as the estimator of $g(\theta)$. We can speak of $t$ as an invariant estimator for a certain class of transformations $g$ if, when the parameter $\theta$ is transformed by $g$ to $g(\theta)$, the estimator $t$ is transformed to $g(t)$ \[10:233\].

If the statistician is working with an invariant estimator where the estimate of $\theta$ is $t$, then he can conclude that his estimate for $\theta + k$ is $t + k$ and his estimate for $k\theta$ is $kt$ \[10:233\].

Thus, the property of invariance permits the transformation of a parameter to be translated into the transformation of its estimator.

**Summary.** Three desirable properties of an estimator are consistency, efficiency, and invariance. Unbiasedness is desirable when the estimator is used in repeated sampling from the same population. Unbiasedness means that, on the average, the estimator equals the parameter being estimated. Consistency means that as the sample size increases, the estimator will more closely approximate the true parameter value. Efficiency is a comparative measure between estimators where the estimator with the smaller mean square error is more efficient. Finally, invariance means that if a transformation operation is performed on a parameter, the identical transformation can be performed on the estimator resulting in the transformed estimator becoming a valid estimator for the transformed parameter. Although these properties are desirable, estimators generally do not possess all of these properties. Therefore, the statisticians must find an estimator with the properties needed for their particular applications.
EMPIRICAL DISTRIBUTION FUNCTION (EDF)

An empirical distribution is a distribution based solely on sample values of a random variable. The empirical distribution can be thought of as an estimation of the true underlying population distribution. The empirical distribution is developed by observing several values of the random variable and constructing a graph $S(x)$ that may be used as an estimate of the entire unknown distribution function $F(x)$ of the random variable (8:59). Conover defines the empirical distribution as follows:

Let $X_1, X_2, \ldots, X_n$ be a random sample. The empirical distribution function $S(x)$ is a function of $x$, which equals the fraction of $X_i$'s that are less than or equal to $x$ for each $x$, $-\infty < x < \infty$ (8:69).

Based on this definition, the graph of the empirical distribution function, $S(x)$, is a step function starting at zero. As each sample value (ordered from lowest to highest) is encountered, a step of height $1/n$ is entered on the graph. This procedure continues until all the sample values have been entered and a height of one has been reached. $S(x)$ resembles a distribution function in that it is a nondecreasing function that goes from zero to one in height. However, $S(x)$ is empirically (from a sample) determined and therefore its name (8:70).

The empirical distribution function is used as an estimator for the population distribution function of the random variable (8:70).

From the empirical distribution function, one can "compute the expectation of the empirical random variable, $E(x)$. We have

$$E(x) = \sum_{i=1}^{n} x_i \frac{1}{n} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

(2.4)

which is just the sample mean, $\bar{x}$ (5:137). Eq (2.4) uses the
discrete random variable form of the expected value definition. Therefore, assuming the empirical distribution acceptably estimates the population distribution leads to the sample mean being an acceptable estimate for the population mean (5:138).

**BEST LINEAR UNBIASED ESTIMATOR (BLUE)**

Knowing what properties are desirable in an estimator still leaves the statistician with the problem of developing an estimator. One estimator is called the best linear unbiased estimation technique. As was mentioned in Chapter I, the BLU estimator is based on order statistics, which is simply an arrangement of random variables in order of magnitude (24:229). A population parameter (θ) can be estimated by a statistic (T) which depends only on the values of n independent random variables: \( x_1, x_2, \ldots, x_n \) (10:265).

The title of this estimator indicates some of the properties that it possesses. Namely, the estimator must be unbiased, 'best', and linear. As was discussed earlier in this chapter, an unbiased estimator has a bias term equal to zero, and, on the average over many trials, the estimator provides estimates equal to the parameter value. Eq 2.1 states the property mathematically. In addition to being unbiased, the BLU estimator must be 'best'. To be best among unbiased estimators, the estimator must have the minimum mean square error (10:265). The mean square error is the sum of the variance term and the square of the bias term (24:267). Since we are dealing with an estimator which is inherently unbiased (i.e., the bias term equals zero), the mean square error simply reduces to the variance term. Therefore, in this case, best implies minimum variance. Finally, the
BLU estimator must be linear. Linearity demands that we consider only "estimators which are linear in the random variables $x_1, \ldots, x_n$", for it is only in comparison with other estimators within this restricted class that we can always find estimators [which are best unbiased] (10:266). Stated mathematically, the estimator appears as follows:

$$T = c_1 x_1 + \ldots + c_n x_n$$  \hspace{1cm} (2.5)

where the coefficients $(c_i)$ must be determined (10:266).

In addition to the properties described above, the best linear unbiased estimator possesses another desirable feature, that of invariance. Mood and Graybill indicate that BLU estimators are a subset of least-squares estimators (25:349). Further, they state that, in general, least square estimators do not possess the invariance property. "There is one important case, however, when the invariant property holds for least-squares estimators, and this is the case of linear functions" (25:350). Therefore, in addition to being unbiased, and possessing minimum variance, the BLU estimator is also invariant.

MINIMUM DISTANCE (MD) ESTIMATOR

Chapter I presented a partial history and description of the minimum distance estimation technique. The efforts of Wolfowitz culminated in his 1957 paper which refined his work toward "developing the minimum distance method for obtaining strongly consistent estimators (i.e., estimators which converge with probability one)" (39:75). In the paper, he emphasized that his method could be used with a variety of distance measuring techniques (39:75). Additionally, Wolfowitz stated
that "it is a problem of great interest to decide which, if any, definition of distance yields estimators preferable in some sense" (39:76). This thesis will in part respond to this challenge, since three distance measures will be used in the minimum distance method for comparison against the best linear unbiased estimation method. The three distance measures to be used are the Kolmogorov, the Anderson-Darling, and the Cramer-von Mises discrepancy measures. Wolfowitz finally summarizes the minimum distance method as follows:

The estimator is chosen to be such a function of the observed chance variables that the d.f. of the observed chance variables (when the estimator is put in place of the parameters and distributions being estimated) is "closest" to the empirical d.f. of the observed chance variables [39:76].

Since 1957, the minimum distance estimation technique has been studied by many other statisticians and has been found to display other desirable estimator properties. The technique has "been considered as a method for deriving robust estimators by Knusel (1969) and Parr and Schucany (1980)" (28:178). Additionally, Parr and Schucany indicate that the method yields "strongly consistent estimators with excellent robustness properties" (27:5) when used to estimate the location parameter of symmetric distributions (27:5). They define robust estimation as "efficient or nearly efficient (at a model) estimation procedures which also perform well under moderate deviations from that model" (27:2). They attempt to explain why the minimum distance estimator possesses robustness properties:

It may well be inquired as to why an estimator obtained by minimization of a discrepancy measure which is useful for goodness-of-fit purposes (and, hence, in many cases extremely sensitive to outliers or general discrepancies from the model) should be hoped to possess any desirable 'robustness' properties. 'It turns out that, in most
cases... while the discrepancy measure itself may be fairly sensitive to the presence of outliers, the value... which minimizes the discrepancy... is much less so [27:5-6].

Finally, they state that the method presents a trade-off between efficiency considerations and robustness considerations (28:179).

In addition to consistency, robustness and efficiency, investigators have revealed other attractive features of the minimum distance estimation technique. Parr and Schucany indicate that "minimum distance estimators share an invariance property with maximum likelihood estimators... It operates in a manner analogous to maximum likelihood methods in simply selecting a 'best approximating distribution' from those in the model" (27:9). Additionally, Parr states that the method is very easy to implement. "Given a set of data, a parametric model, and a distance measure between distribution functions, all that is needed is an omnibus minimization routine to compute the estimator" (26:1207-1208). Finally, minimum distance estimators provide meaningful results even if the conjectured parametric model is incorrect. MD-estimation still provides the best approximation in terms of probability units with regard to the conjectured distribution (26:1208). "This is a feature not enjoyed by other estimation methods such as the maximum likelihood" (26:1208). Therefore, MD-estimation can be a very useful tool for the statistician.

The minimum distance estimation technique uses a distance measure and, for this reason, is closely linked with certain goodness-of-fit tests. As explained by Stephens, goodness-of-fit statistics are "based on a comparison of F(x) with the empirical distribution function F_n(x)" (35:730). In a goodness-of-fit test, one is interested in fitting an empirical distribution function, described
earlier, with a fully specified (i.e., with known parameters) distribution function. The test for whether the fit is 'good' is normally a measure of distance between the two distribution curves. In contrast, minimum distance estimation uses a parent distribution family with certain unknown parameters. The estimates of the unknown parameters are those parameter values which minimize the distance measure between the empirical distribution and the parent distribution being investigated. The three distance measures to be used in this study are described next.

**Kolmogorov Distance.** The statistic suggested by Kolmogorov in 1933 is the largest absolute distance between the graphs of the empirical distribution function, \( S(x) \), and the hypothesized distribution function, \( F(x; \theta) \) measured in the vertical direction (8.345). Symbolically, the Kolmogorov distance (\( D \)) is given by:

\[
D = \sup x \left| F(x; \theta) - S(x) \right|
\]

which reads \( D \) equals "the supremum, over all \( x \), of the absolute value of the difference \( F(x; \theta) - S(x) \) " (8.347). Stephens provides a computational form for all of the distance measures to be used in this study where he lets \( z_1 = F(x_1) \), \( i = 1, 2, \ldots, n \). For the Kolmogorov distance, the computational form is as follows:

\[
D^+ = \max_{1 \leq i \leq n} \left( \frac{1}{n} - z_1 \right)
\]

\[
D^- = \max_{1 \leq i \leq n} \left[ z_1 - \frac{(i-1)}{n} \right]
\]

\[
D = \max (D^+, D^-)
\]

(35:731). These computational formulae provide the maximum distance
between the empirical distribution function, which is a step function, and the conjectured distribution function, F(x; θ).

Cramer-von Mises Distance. The Cramer-von Mises statistic is actually a member of the Cramer-von Mises family of distance measures which is "based on the squared integral of the difference between the EDF and the distribution tested:

\[ W^2 = \int_{-\infty}^{\infty} (F_n(x) - F(x; \theta))^2 \beta(x) \, dx \]  

(2.8)

The function \( \beta(x) \) gives a weighting to the squared difference" (34:2). The Cramer-von Mises statistic is produced by setting the weighting function equal to one, \( \beta(x) = 1 \) (34:2). The computational form of the Cramer-von Mises statistic is given by Stephens as follows:

\[ W^2 = \sum_{i=1}^{n} [z_i - (2i - 1)/2n]^2 + (1/12n) \]

(2.9)

(35:73i). This formula uses the same symbolism as the computational form of the Kolmogorov distance measure.

Anderson-Darling Distance. The Anderson-Darling distance measure is actually another member of the Cramer-von Mises family. In this case, however, the weighting factor is \( 1/u(1-u) \) where \( 0 < u < 1 \) (27:4). "This weight function counteracts the fact that the discrepancy [in Eq 2.9] between \( F_n(x) \) and \( F(x; \theta) \) is necessarily becoming smaller in the tails, since both approach 0 and 1 at the extremes (34:2). Therefore, the Anderson-Darling weighting function gives "greater importance to observations in the tail than do most of
the EDF statistics" (34:2). Stephens gives the computational form of 
the Anderson-Darling statistic as follows:

\[ A^2 = - \{ \sum_{i=1}^{n} (2i - 1) \left[ \ln z_i + \ln (1 - z_{n+1-i}) \right] \} / n - n \]  

(35:731). Again, this computational formula uses the same symbology
used for the other two distance measures' computational formulae.
III. Pareto Distribution

This chapter will first relate the history of the Pareto distribution. A summary of various socio-economic and military applications will follow this historical perspective. Then a detailed description of the Pareto function will be presented. Finally, this chapter will describe the best linear unbiased and the minimum distance estimation techniques as applied specifically to the Pareto function.

HISTORY

In 1897 Vilfredo Pareto (1848-1923), an Italian-born Swiss professor of economics, formulated an empirical law which bears his name (16:233). Pareto's Law was based on his study of the distribution of incomes in several European countries during the nineteenth century. The mathematical results of the study were summarized as follows:

\[ N = Ax^{-c} \]  

(3.1)

where \( N \) is the number of people having incomes equal to or greater than income level \( x \). \( A \) and \( c \) are parameters where \( c \) is sometimes referred to as Pareto's constant or the shape parameter (16:233). Pigou summarized Pareto's findings in the following statement:

It is shown that, if \( x \) signify a given income and \( N \) the number of persons with incomes exceeding \( x \), and if a curve be drawn, of which the ordinates are logarithms of \( x \) and the abscissae logarithms of \( N \), this curve, for all the countries examined, is approximately a straight line, and is, furthermore, inclined to the vertical axis at an angle, which, in no country, differs by more than three or four degrees from 56°. This means (since \( \tan 56^\circ = 1.5 \)) that, if the number of incomes greater than \( x \) is equal to \( N \), the number greater than \( mx \) is equal to \( N(1/m)^{1.5} \),
whatever the value of \( m \) may be. Thus the scheme of income distribution is everywhere the same [29:647].

The Pareto premise, then, as deduced from his mathematical findings and stated in economic rather than mathematical terms is as follows:

Hence, what this thesis amounts to in effect is that, on the one hand, anything that increases the national dividend must, in general, increase also the absolute share of the poor, and, on the other hand--and this is the side of it that is relevant here--that it is impossible for the absolute share of the poor to be increased by any cause which does not at the same time increase the national dividend as a whole. . . we cannot be confronted with any proposal the adoption of which would both make the dividend larger and the absolute share of the poor smaller, or 'vice versa' [29:648].

Pareto felt, therefore that his law was 'universal and inevitable--regardless of taxation and social and political conditions' [16:233].

Since the statement of Pareto's Law, several renowned economists have refuted the law's sweeping applicability (16:233). In particular, Pigou identified defects in its statistical basis, arguing that the differences in inclination of the plotted lines were significant. Additionally, he argues that such a generalization from an empirical study under certain conditions (certain avenues of income such as inheritance and personal effort) cannot justifiably be extended to all social conditions (29:649-655).

The general defence of "Pareto's Law" as a law of even limited necessity rapidly crumbles. His statistics warrant no inference as to the effect on distribution of the introduction of any cause that is not already present in approximately equivalent form in at least one of the communities--and they are very limited in range--from which these statistics are drawn. This consideration is really fatal; and Pareto is driven, in effect, to abandon the whole claim . . . [29:654-655].

Additionally, Champernowne identifies weaknesses in the Pareto Law.
He indicates that the use of the Pareto constant as a measure of income distribution inequality between communities suffers from two problems. Firstly, the measure only addresses income before taxation. Secondly, the measure only applied to income distributions among the rich and breaks down when applied to those with medium incomes (7:609).

Finally, Fisk discusses the value of the Pareto distribution regarding its ability to describe distributions of income. He states that the "Pareto curve fits income distributions at the extremities of the income range but provides a poor fit over the whole income range" (12:171).

Therefore, Pareto's Law with regard to income distributions is no longer highly touted. However, other disciplines have found application of the Pareto distribution to be very useful.

APPLICATIONS

Socio-economic Related Applications. Although the Pareto distribution was formulated as a reflection of income distribution, the Pareto distribution has proven to be useful in many other areas of investigation. Johnson and Kotz indicate the Pareto distribution can be useful in describing many socio-economic or naturally occurring quantities. Examples include the distributions of city population sizes, fluctuations in the stock market, and the occurrence of natural resources. The Pareto is useful in these areas because they often display statistical distributions with very long right tails (16:242).

Koutrouvelis listed some additional areas where the Pareto distribution had successfully been used. These areas include: business mortality rates, worker migration, property values and inheritance, and
Johnson and Kotz additionally identified the area of personal income investigation as an area where the Pareto distribution was applicable. In 1982, Wong used the Pareto in his analysis of income. He indicates that many individuals underreport their true incomes to avoid a portion of their tax payments. Wong shows the applicability of the Pareto in reflecting this underreporting phenomena.

**Militarily Related Applications.** In addition to socio-economic interests, the Pareto distribution has proven useful in many areas of interest to the military. These areas include fallout mass-size distributions, interarrival time distributions, and failure time distributions. This section will address each of these areas in turn.

E. C. Freiling conducted a study for the U.S. Naval Radiological Defense Laboratory concerning a comparison of distribution types for describing "the size distribution of particle mass in the fallout from land-surface bursts". In this study, he compared the lognormal distribution with the Pareto. He determined that with the effects of the uncertainties playing in the problem, the differences in descriptive ability of the two distributions were trivial. He indicated that the lognormal "has the esthetic advantage of an observationally confirmed theoretical basis in the case of airburst debris". However, if truncation is required, the Pareto distribution has "the practical advantage of simplifying further calculations of particle surface distribution".

A Pareto description of interarrival times has played an important part in two other studies, one involving interarrival times in general...
and the second involving telephone circuit error clustering. Bell, Ahmad, Park and Lui performed the general interarrival time study supported by a grant from the Office of Naval Research. They indicate that interarrival time distributions are usually thick-tailed as compared to Gaussian or Poisson processes for like distributions. They state that the Pareto can provide a variety of tail thicknesses depending on the value of the shape parameter employed (2:1). In the telephone circuit paper, Berger and Mandelbrot propose a new model to describe error occurrence on telephone lines. They conclude that the Pareto distribution can well be used to approximate the distribution of inter-error intervals.

Finally, the Pareto distribution has proven useful in life testing and replacement policy situations. Davis and Feldstein show the Pareto as a competitor to the Weibull distribution with regard to time to failure of a system since, "unlike the Weibull, it does not give rise to infinite hazard at the origin nor hazard increasing without bound" (9:306). Kaminsky and Nelson illustrate the use of the Pareto in developing replacement policy. The Pareto can be used to predict component replacement times based on an accumulation of early failure data (17:145).

**Pareto Function**

The mathematical formulation of Pareto's Law on income distribution is shown in Eq (3.1). This law corresponds to the following Pareto probability density function as given by Johnson and Kotz:

\[ P(x) = \Pr[X \geq x] = \left(\frac{a}{x}\right)^c \quad a > 0, \ c > 0, \ x > a \quad (3.2) \]
In this equation \( P(x) \) gives the probability that income is equal to or greater than \( x \), while \( a \) corresponds to some minimum income (16:234).

The cumulative distribution function (cdf) of \( X \) resulting from Eq (3.2) gives the following Pareto distribution:

\[
F_X(x) = 1 - \left(\frac{a}{x}\right)^c \quad a>0, \ c>0, \ x>a
\]  

(16:234). During Mandelbrot's investigation concerning the Pareto distribution, he distinguishes between two forms of the Pareto Law: the Strong Law of Pareto and the Weak or Asymptotic form of the Law of Pareto. Mandelbrot's Strong Law of Pareto is of the form shown in Eq (3.3) and is written as follows:

\[
1 - F_X(x) = \left(\frac{x}{a}\right)^{-c} \quad x>a
\]  

(3.4)

Mandelbrot's Weak or Asymptotic form of the Pareto Law is written as follows:

\[
1 - F_X(x) \sim \left(\frac{x}{a}\right)^{-c} \quad \text{as} \quad x \to \infty
\]  

(3.5)

The Weak form implies that if the log of the left side of the relation is graphed against log \( x \) "the resulting curve should be asymptotic to a straight line with slope equal to \(-c\) as \( x \) approaches infinity" (16:245).

Grouping Pareto Distributions by Kind. There are several versions of the Pareto cumulative distribution function. Often, these versions are grouped according to 'kind'. There are three labels used in this type of grouping scheme: Pareto distributions of the first kind, of the
second kind, and of the third kind.

A distribution of the form shown in Eq (3.3) is referred to as a Pareto distribution of the first kind (16:234). A Pareto distribution of the second kind is written as follows:

\[ F(x) = 1 - \frac{K}{(x + C)^C} \quad (3.6) \]

(16:234). This form differs from the Pareto distribution of the first kind through the addition of another quantity, C, in the denominator of the second term on the right hand side of the equation.

In addition to the two distribution kinds above, Pareto suggested a third law, the distribution of which Mandelbrot calls a Pareto distribution of the third kind. The mathematical form is as follows:

\[ F(x) = 1 - \frac{k_2 e^{-hx}}{(x + C)^C} \quad (3.7) \]

(16:234). The Pareto distribution of the third kind degenerates to that of the second kind when \( h = 0 \).

**Grouping Pareto Distributions by Parameter Number.** Perhaps a more understandable method of grouping the various forms of the Pareto distribution function is by grouping them according to the number of parameters the form contains. However, before describing these functions, three basic parameters will be defined.

Hastings and Peacock describe three types of parameters which always have a physical or geometrical meaning. These three parameters are those of location (a), scale (b) and shape (c). This study will use this symbology when using these parameters. The location parameter, a, is "the abscissa of a location point (usually the lower or mid point) of
the range of the variate" (15:20). The scale parameter, b, "determines
the scale of measurement of the fractile, x" (15:20). A fractile is a
general element within the range of the variate, X (15:5). Finally, the
shape parameter, c, "determines the shape (in a sense distinct from
location and scale) of the distribution function (and other functions)
within a family of shapes associated with a specified type of variate" (15:20). Using the normal distribution as an example, the mean is the
location parameter because it specifies a kind of mid point for the
distribution. The standard deviation is the scale parameter because it
provides a fractile measurement device for the distribution. "The normal
distribution does not have a shape parameter" (15:20). With this
background on location, scale and shape parameters, we can now proceed
with the discussion on grouping Pareto distributions according to the
number of parameters contained in the distribution expression.

The most commonly used form of the Pareto distribution is the two
parameter form; however, there is a more general form which uses all
three basic parameters of location (a), scale (b), and shape (c). This
section will present this more general form and show how the simpler
forms are derived from it. The three parameter form of the Pareto
distribution is written as follows:

\[ F(x) = 1 - \left[ 1 + \frac{(x-a)}{b} \right]^{-c} \quad x \geq a \quad (3.8) \]

where \( b > 0 \) and \( a > 0 \) (20:218). As stated earlier, the notation of
Hastings and Peacock is used in this equation and in those that follow.

The two parameter Pareto distribution is the most common form of
the distribution and is derived from Eq (3.8) by eliminating either the
location or the scale parameter from the equation. One way to obtain a two parameter distribution function is to set the location parameter equal to zero. For \( a=0 \) we obtain a Pareto distribution of the second kind as shown in Eq (3.6) where \( K=b^C \) and \( C=b \). This special case is sometimes referred to as the Lomax distribution (Z1:Z18). Another method of effectively eliminating one of the parameters is to set the location parameter equal to the scale parameter. Setting \( a=b \) in Eq (3.3) results in the usual formulation of the Pareto distribution and is the Pareto distribution of the first kind as shown in Eq (3.3).

The simplest form of the Pareto distribution is the one parameter version which can be obtained by setting both the location and the scale parameter equal to one. Setting \( a=b=1 \) in Eq (3.8), the following distribution function results:

\[
F(x) = 1 - x^{-C} \quad x \geq 1 \quad (3.9)
\]

This one parameter form is regarded as the 'standard form' of the Pareto distribution (16:240).

Since most of the many versions of the Pareto distribution can be derived from the more general three parameter model, this thesis investigates the three parameter distribution. This should ensure that results of this study can be used in a wider variety of applications where estimation is required.

**PARAMETER ESTIMATION**

This section describes the estimation methods used in this study as applied specifically to the Pareto distribution. First the best linear unbiased estimators are presented along with the procedure used to
transform these estimators into a computational form. Then the minimum
distance estimation formulas will be adapted to the Pareto distribution.

*Best Linear Unbiased Estimator.* As was mentioned by Kulldorff and
Vännman, the general, three parameter form of the Pareto cumulative
distribution function has received little attention from statisticians
working on the development of estimators (20:218). Hence, many
estimators have been developed for special cases of the two parameter
formulation while few estimators are available for study of the more
general distribution form.

Kulldorff and Vännman successfully derived BLU estimators for three
cases of the general Pareto distribution where the shape parameter is
always assumed to be greater than two. Specifically, these cases are:
- scale parameter when the location and shape are known;
- location parameter when the scale and shape are known; and
- location and scale parameters when the shape is known (20:218-224). The estimators
developed for the third case are the estimators used in this study,
since only shape parameters will be explicitly specified for the Pareto
distribution being investigated. However, these estimators are useful
only when $c > 2$.

Vännman later presented the BLU estimators for the same three
cases shown above with the condition that the shape parameter is equal
to or less than two (36:704). Therefore, his estimators will be used
for the cases when $c \leq 2$.

**BLUEs for Shape Greater Than 2.** As stated earlier,
Kulldorff and Vännman developed BLU estimators for both the location
and scale parameters with the shape parameter known and greater than
two. From Chapter II, we recall that the BLU estimator is based on
order statistics where the random variables are arranged in order of magnitude from smallest to largest (Z4:229). Therefore, the elements of the drawn sample are ordered from smallest to largest to provide the order statistics where $x_{(1)} \leq x_{(2)} \leq \ldots \leq x_{(n)}$. Here $x_{(1)}$ is the smallest valued observation and $x_{(n)}$ is the largest valued observation from the sample of size $n$. Since a BLU estimator must take the form of a linear combination of the ordered random variables, the BLU estimators for the location and scale parameters of the Pareto distribution with specified shape must be a linear combination of the ordered sample observations, where the coefficients of these observations are to be determined. In developing their BLU estimators, Kulldorff and Vännman derived this linear relationship and determined the coefficients which are based on the sample size and the specified shape parameter. The BLU estimators for location, $a$, and scale, $b$, are written as follows:

$$\hat{a} = x_{(1)} - \frac{Y}{(nc-1)(nc-2) - ncD}$$ \hspace{1cm} (3.10)$$

$$\hat{b} = \frac{Y(nc-1)}{(nc-1)(nc-2) - ncD}$$

$$= \frac{(nc-1)[x_{(1)} - \hat{a}]}{(nc-1)}$$ \hspace{1cm} (3.11)$$

The authors note that in the specific case when $a=b$, the BLU estimator reduces to the following:

$$\hat{a} = \frac{1}{1 - (1/nc)} x_{(1)}$$ \hspace{1cm} (3.12)$$

Equations (3.10) and (3.11) both contain two quantities, $Y$ and $D$, which still need to be defined. $Y$ is defined in terms of $D$ and an
additional new term $B_i$, while $D$ simply contains the new term $B_1$. $B_i$ is defined in terms of the sample size, $n$, and the specified shape parameter, $c$. Therefore, by computing the $B_i$ terms, both $D$ and $Y$ can be determined. With $D$ and $Y$ known, one can then calculate the BLU estimators of location and scale. The expressions for $Y$, $D$ and $B_i$ are as follows:

$$Y = (c+1) \sum_{i=1}^{n-1} B_i x_i(n) + (c-1)B x(n) - D x(1) \quad (3.13)$$

$$D = (c+1) \sum_{i=1}^{n-1} B_i + (c-1)B(n) \quad (3.14)$$

$$B_i = \frac{\Gamma(n-1+1) \Gamma(n+2/c)}{\Gamma(n-i+1-2/c) \Gamma(n+1)} \quad i = 1, 2, \ldots, n \quad (3.15)$$

Equations (3.10) to (3.15) are the Kulldorff and Vannman equations (20:219-223).

To obtain the BLU estimators, we must calculate all of the $B_i$ values for $i = 1, 2, \ldots, n$. Eq (3.15) shows that $B_i$ contains four gamma functions which would require considerable computational time; however, the expression can be simplified to reduce the computational load.

Banks and Carson indicate that "the gamma function can be thought of as a generalization of the factorial notion which applies to all positive numbers, not just integers" (1:144). They show that for any positive real number, $p$, the gamma function of $p$ is as follows:
\[ \Gamma (p) = (p-1) \Gamma (p-1) \quad (3.16) \]

Since \( \Gamma (1) = 1 \), we see that if \( p \) is an integer than Eq (3.16) reduces to (1:144):

\[ \Gamma (p) = (p-1)! \quad (3.17) \]

Eq (3.16) and Eq (3.17) will be used to simplify the \( B_1 \) term to a more manageable computational form. The first gamma function in the numerator and the last gamma function in the denominator will always be gamma functions of integer values; therefore, Eq (3.17) can be used to transform these terms to common factorial terms. Eq (3.16) will be used on the remaining gamma function in the numerator to assist in the reduction process. Simplification of the first two \( B_1 \) terms (30) will reveal a pattern which will simplify the evaluation process:

\[
B_1 = \frac{\Gamma (n+1) \Gamma (n+2/c)}{\Gamma (n+1-2/c) \Gamma (n+1)}
\]

\[
= \frac{\Gamma (n+1) \Gamma (n+2/c)}{\Gamma (n+2/c) \Gamma (n+1)}
\]

\[
= \frac{(n-1)! \Gamma (n-2/c)}{n (n-1)! \Gamma (n-2/c)}
\]

\[
= \frac{(n - 2/c) / n}{1 - 2/(cn)} \quad (3.18)
\]
Solving for $B_2$ in a similar manner yields the following:

$$B_2 = \frac{\Gamma(n-2+1) \Gamma(n+1-2/c)}{\Gamma(n-2+1-2/c) \Gamma(n+1)}$$

$$= \frac{\Gamma(n-1) \Gamma(n+1-2/c)}{\Gamma(n-1-2/c) \Gamma(n+1)}$$

$$= \frac{(n-2)! (n-2/c) \Gamma(n-2/c)}{n! \Gamma(n-1-2/c)}$$

$$= \frac{(n-2)! (n-2/c) (n-1-2/c) \Gamma(n-1-2/c)}{n (n-1)(n-2)! \Gamma(n-1-2/c)}$$

$$= \frac{(n-2/c) (n-1-2/c)}{n (n-1)}$$

$$= [1 - 2/(cn)] [1 - 2/c(n-1)]$$  \hspace{1cm} (3.19)

Equations (3.18) and (3.19) reveal the following pattern for the $B_n$.

$$B_n = [1 - 2/cn] [1 - 2/c(n-1)] \ldots [1 - 2/c(1)]$$  \hspace{1cm} (3.20)

The following notation will allow even further simplification of the $B$ value computations:
Let $t_1 = \frac{2}{c(n)}$, $t_2 = \frac{2}{c(n-1)}$, ..., $t_n = \frac{2}{c(1)}$.

Let $u_1 = 1 - t_1$, $u_2 = 1 - t_2$, ..., $u_n = 1 - t_n$.

Then $B_1 = u_1$, $B_2 = u_1u_2$, ..., $B_n = u_1u_2...u_n$.

And in general, the computational form is as follows:

$$B_i = \prod_{j=1}^{i} u_j$$  \hspace{1cm} (3.21)

where $u_j = 1 - t_j$ and $t_j = \frac{2}{c(n-j+1)}$ for $j = 1, 2, ..., i$

(30). Equipped with these relations, we can now write the following recursive relationship which will allow simpler calculations as recommended by Vännman (36:705):

$$B_i = (1 - \frac{2}{c(n-i+1)}) B_{i-1} \hspace{1cm} i = 1, 2, ..., n$$  \hspace{1cm} (3.22)

With these relationships available, the programming of these calculations will be much simpler.

**BLUEs for Shape Equal to or Less Than 2.** Vännman indicates that the variance of the Pareto distribution does not exist when the shape parameter, $c$, is equal to or less than 2; therefore, the above formulas for BLU estimators of location and scale do not apply. He further states, however, that if only the first $k$ order statistics are used in the estimator, where $2 \leq k \leq \left(n+1 - \frac{2}{c}\right)$, then the variance of the estimator does exist, with the added condition that the shape parameter satisfies the following relationship: $\frac{2}{n} \leq c \leq 2$. He indicates that the most efficient estimator is obtained by basing the estimator on the first $k$ order statistics where $k = n - \lceil \frac{2}{c} \rceil$. In this equation the bracketed fraction implies that only the integer
portion of the fraction is used in the calculation \((36:705-707)\). The formulas for the location and scale parameters based on the first \(k\) order statistics are as follows \((36:707)\):

\[
a_k = x_{(1)} - \frac{b_k}{(nc-1)} \quad \text{(3.23)}
\]

and

\[
b_k = \frac{k-1}{U_k} \left( \frac{c+1}{k} \right) \sum_{i=1}^{k} B_i x_{(i)}
+ \left( \frac{(n-k+1)c - 1}{k} \right) B_k x_{(k)}
- \left( \frac{(nc-1)/(nc)}{(nc-2 - U_k)} \right) x_{(1)} \quad \text{(3.24)}
\]

where

\[
U_k = \frac{(nc-2)(nc-c-2) - nc\left(n-k\right)c - 2B_k}{(nc-1)(c+2)} \quad \text{(3.25)}
\]

Again, Equations (3.23) and (3.24) can only be used where \(k\) represents the first \(k\) order statistics and where \(k \leq n + 1 - 2/c \). To obtain the most efficient BLU estimator, Vanman indicates that \(k\) should additionally satisfy the following: \(k = n - \lceil 2/c \rceil \). He further states that in the case where \(2/c\) is already an integer value, then eq (3.24) simplifies to the following \((36:707)\):

\[
b = \frac{(c+1)(c+2)(nc-1)}{(nc-2)(nc-c-2)} \left\{ \frac{n-2/c}{\sum_{i=1}^{n} B_i x_{(i)} - \left( (nc-2)/(c+2) \right) x_{(1)} } \right\} \quad \text{(3.26)}
\]

Eq (3.26) can then be entered into Eq (3.23) to obtain the BLU estimator for location. However, to use eq (3.26), the simplified version of the BLU estimator of the scale parameter, four conditions must exist:
1) The shape parameter, $c$, must be specified.

2) $2/n < c \leq 2$

3) $2/c$ is an integer

4) $2 \leq k = n - 2/c$

Finally, Vannman notes two simplified expressions for $B_1$ when the shape parameter equals 1 or 2. He indicates that if $c = 1$, then $B_1 = (1 - i/n) \{ 1 - (i-1)/n \}$. If $c = 2$, then $B_1 = 1 - i/n$ (36:705). The computer program verification and validation phase of this research revealed that Vannman's simplified expression for $B$ when $c = 1$ was incorrect in the published reference. By setting $c = 1$ and simplifying Eq (3.15), the error in Vannman's published formula was found. To generate correct $B$ values with $c = 1$, Vannman's bracketed term $\{ 1 - (i-1)/n \}$ must be changed to $\{ 1 - i/(n-1) \}$. These simplified expressions will be valuable in the computer programming phase of this study, since $B$ values must be calculated to determine the BLU estimates.

**Minimum Distance Estimator.** The general computational forms of the three distance measures used in this study are presented in Chapter II and are reflected in equations (2.7), (2.9), and (2.10). To apply these measures using a Pareto distribution, we simply substitute our hypothesized Pareto distribution function, $P_1$, for the $z_1$ value currently shown in these equations, where the starting point for the estimates of location and scale will be the BLU estimates. This hypothesized Pareto cdf can be written as follows:

$$P_1 = F(x_1; \hat{a}, \hat{b}, \hat{c}) = 1 - (1 + (x_1 - \hat{a})/\hat{b})^{-\hat{c}}$$ (3.27)
The minimization routine, ZXMIN, from the International Mathematical Statistics Library (IMSL) will then alter the values of location and scale to obtain the minimum distance measure values. These altered estimates for location and scale then become the minimum distance estimates for that particular distance measure. The procedural details are covered in more depth in the following chapter.
IV. Monte Carlo Analysis

This chapter will describe the specific analysis tool used in this study to compare the best linear unbiased and the minimum distance estimation techniques. The tool is called Monte Carlo analysis. Following a general discussion of the Monte Carlo method, the specific application of the method in this study will be described. This application description will present the three step process of Monte Carlo analysis along with the detailed procedures involved within each step.

MONTE CARLO METHOD

The Monte Carlo method, or the method of statistical trials (6:1), falls within the realm of experimental mathematics. Hammersley and Handscomb indicate that the essential difference between theoretical and experimental mathematicians "is that theoreticians deduce conclusions from postulates, whereas experimentalists infer conclusions from observations" (13:1). Monte Carlo analysis is a member of the experimental mathematics branch since it deals with mathematical experiments on random numbers (13:2). A further explanation of the Monte Carlo method is provided by Schreider:

The Monte Carlo method (or the method of statistical trials) consists of solving various problems of computational mathematics by means of construction of some random process for each such problem, with the parameters of the process equal to the required quantities of the problem. These quantities are then determined approximately by means of observations of the random process and the computation of its statistical characteristics, which are approximately equal to the required parameters (6:1).
This description of the Monte Carlo method reflects how well suited the method is for this particular study, since the description mirrors the process used to compare the two estimation techniques.

**MONTE CARLO STEPS AND PROCEDURES**

This study uses a three step Monte Carlo process to compare best linear unbiased estimation with minimum distance estimation (using three distinct distance measures) as applied to the Pareto distribution. First, one generates random variates from a specified Pareto distribution (i.e., a Pareto distribution with known parameters). Second, the two estimation techniques are used to obtain parameter estimates based on the random sample data from the first step. Third, the resulting estimates are compared to determine which estimation technique provided the better parameter estimates (p.27).

**Step 1: Data Generation.** Using the Monte Carlo technique, we generate our own random data using the random number generator of the VAX 11/785 (VMS) computer system located at the Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio. A random number generator generates random numbers uniformly distributed on \([0,1]\) (p.293). Parr stated that there were four items required to perform a minimum distance estimation: a set of data, a parametric model, a distance measure, and a minimization routine (p.1207-1208). The data generation step supplies the first two items by generating the data based on a specified parametric model, the Pareto distribution.

In the first step, the researcher generates the random sample data needed to create the controlled environment, using different parameter values for each data set. To evaluate the effect of sample size on the
estimators and ensure validity, sample sizes (n) of 6, 9, 12, 15, and 18 are used. Additionally, shape parameters (c) of 1.0, 2.0, 3.0, and 4.0 are used with the location parameter (a) set to 1 and the scale parameter (b) set to 1 for each sample size resulting in 20 total data sets. The random sample data required for the study are random variates from a specified Pareto distribution. Previous thesis students had used distributions for which computer programs were already available to generate random variates using subroutines from the International Mathematical Statistics Library (IMSL) (4:27; 18:43). However, IMSL does not contain a similar subroutine for the Pareto distribution. Therefore, the random variate relationship was derived using the inverse transform technique (1:294-295) on the general three parameter Pareto distribution function shown in Eq (3.8) with location parameter of 1 and scale parameter of 1. The derivation of the Pareto random variate relationship begins by substituting \( a=1 \) and \( b=1 \) into Eq (3.8) which yields the following:

\[
F(x) = 1 - \left( \frac{1}{x} \right)^c
\]  

(4.1)

Letting \( R \) be a random number between 0 and 1 and letting \( X \) be the random variate, we have:

\[
R = 1 - \left( \frac{1}{X} \right)^c
\]

(4.2)

Solving for \( X \) yields the Pareto random variate relationship:

\[
X = \left( \frac{1}{R} \right)^{1/c}
\]

(4.3)

For each of the 20 data sets, 1000 samples are generated where each
A data set is characterized by a unique sample size \((n = 6, 9, 12, 15, \text{ or } 18)\) and shape parameter \((c = 1.0, 2.0, 3.0, \text{ or } 4.0)\) with location parameter and scale parameter set equal to 1. Therefore, a total of 20000 random sample sets are generated, since 20 separate data sets are required to reflect all the combinations of sample sizes and shape parameters. Previous studies also used 1000 samples to evaluate the estimation techniques (4:28; 18:43). A computer subroutine, PARVAR, was written to generate the 20000 random sample sets from the three parameter Pareto distribution. The IMSL subroutine VSRTA was used on each sample set of size \(n\) to arrange the random variates from smallest to largest. The output was then used by each of the estimation technique subroutines.

**Step 2: Estimate Computation.** The second step of the Monte Carlo process is to use both of the estimation techniques, best linear unbiased and minimum distance estimation, to compute estimates based on the random sample data sets. We first present the procedures used for finding the best linear unbiased estimates. This presentation is followed by the minimum distance estimation procedures.

Using each of the data sets along with the best linear unbiased estimators for the location and scale parameters of the Pareto distribution function for each data set, one obtains 1000 best linear unbiased estimates for the parameters of each particular Pareto distribution sampled. The computer subroutines written to perform this task were titled BLCGT2 and BLCLE2. These subroutines were eventually run against all 20 data groups.

The minimum distance estimation process develops six minimum distance estimators using the 'BLUE' estimates of location and scale.
for each sample of size n as the starting values for the hypothesized
distribution function, \( F(x; \hat{a}, \hat{b}, \hat{c}) \), which in our computational notation
is equal to \( \hat{z}_1 \). The IMSL minimization subroutine, ZXMIN, then
minimizes the computational form of each distance measure in turn. For
instance, by varying the value of the location parameter while holding
the scale equal to the BLUE for scale, ZXMIN finds the value of the
location parameter which minimizes the distance between the hypothesized
distribution and the empirical distribution function for each sample of
size n. This new value for the location parameter is the single
parameter minimum distance estimate of the location parameter.
Alternatively, by holding the location parameter equal to the BLUE for
location, ZXMIN uses the same procedures to obtain a single parameter
minimum distance estimate for the scale parameter. Finally, ZXMIN finds
what we call a double parameter minimum distance estimate by varying
both the location and the scale parameters in the same minimization
calculation. The result of a double parameter minimum distance estimate
run is a simultaneous estimate of both location and scale. The two
single parameter minimization techniques (i.e. one for location and one
for scale) along with the double parameter minimization technique are
applied to each of the three distance measures, resulting in 12 minimum
distance estimates for each data set generated. The computer
subroutines written to perform these tasks are KSMD, KSAMD, and KSBMD
for the Kolmogorov distance measure. For the Cramer-von Mises distance
measure, the subroutines CVMD, CVAMD, and CVBMD were written. Finally,
for the Anderson-Darling distance measure, the subroutines written are
entitled ADMD, AOBMD, ADABMD. The source code for these subroutines is
located in Appendix B. Each of these subroutines is run against all 20
Step 3: Estimate Comparison. The third and final step in the Monte Carlo analysis is estimate comparison. In this step, the mean square error (MSE) approach is used to evaluate which estimation technique provides more accurate parameter estimates (4:31).

Many statisticians support the use of MSE as a good evaluation for comparing estimators. Mendenhall and Scheaffer state that MSE is the expected value of $(\hat{\theta} - \theta)^2$. They further indicate that the mean square error can be written as the summation of the variance and the square of the bias of an estimator (24:267). Since we seek unbiased and relatively efficient estimators, small MSE values should provide a good indication of estimators possessing these two desirable properties and should therefore provide a good estimator comparison tool. Mendenhall further describes a method for evaluating an estimator which parallels the method used in this study:

Thus the goodness of a particular estimator could be evaluated by testing it by repeatedly sampling a very large number of times from a population where the parameters were known and obtaining a distribution of estimates about the true value of the parameter. This distribution of estimates would be referred to as the 'distribution of the estimator'. Those estimators possessing distributions that grouped most closely about the parameter would be regarded as 'best'. Hence, the relative 'goodness' of estimators may be evaluated by comparing their biases and their variances (23:14-15).

Since the MSE is a function of both the variance and the bias, an MSE comparison should reflect the goodness of the estimators considered, as suggested by Mendenhall. However, Mood and Graybill warn that "except in trivial cases, there is no estimator whose mean-squared error is a minimum for all values of $\theta$" (25:167). That is, for a given $\theta$ value, estimator A may produce the smallest MSE, while for another value of the parameter, estimator B may provide the smallest MSE. However, Mood
and Graybill do concede that the MSE does provide a useful guide for estimator comparisons. In fact, they do end up using the MSE as their guide in searching for minimum risk estimators. Finally, Liebelt provides two reasons why minimizing the average mean square error is a credible criteria for evaluating good estimators:

First, if the mean square error is zero or near zero then the dispersion of the estimate from the true value is also zero or near zero. Secondly, the choice of minimizing the average mean square error is an easy mathematical procedure, whereas other choices often lead to insurmountable analytical difficulty [21:137].

Therefore, many authors support the use of comparative mean square errors as a valid technique for evaluating the relative worth of estimators, where the estimator with the smallest MSE is considered the 'best' estimator for a given set of parameter values.

The term mean square error is very descriptive of the procedures used during the evaluation. The 'error' from each of the 1000 samples of size n is found by subtracting the estimated parameter from the true population parameter. This error term is then squared, giving the 'square error.' Finally, the mean of the 1000 'square error' terms is found by summing these terms and dividing by 1000, thereby producing a 'mean square error' (4:32). The estimator providing the smallest MSE, therefore, is the best estimation technique to use. The formula for calculating the MSE is as follows:

\[
\text{MSE}(\hat{\theta}) = \frac{1}{N} \sum_{i=1}^{N} (\hat{\theta}_i - \theta)^2 / N
\]  

(4.4)

"where \( \theta \) is the true value of the parameter, \( \hat{\theta}_i \) is the \( i \)th estimate, and \( N \) is the number of times the estimation is performed--in this..."
analysis, \( N = 1000 \) \( (4:32) \). In this case, the parameters being evaluated are the location and scale parameters. Of course, the computer was used to perform the MSE calculations because of the large number of calculations involved. The MSE calculations are embedded in the main program, BLUMD, thereby eliminating the need to store large numbers of variate and estimate values. The MSE calculations result in seven estimation error indicators for the location parameter and seven estimation error indicators for the scale parameter for every specified Pareto distribution considered. The seven estimation error indicators for each parameter correspond to the seven estimation techniques used: the best linear unbiased estimator and the six minimum distance estimators. The estimation technique which reflects the smallest MSE is considered the best parameter estimation technique for that specified Pareto distribution.

The subroutines described above in the three step Monte Carlo analysis were merged into a computer program (BLUMD) which output the MSE values for the estimation techniques being compared. The source code is found in Appendix B. The logical steps or pseudocode for the program is listed in Figure 1.

Each of the subroutines in BLUMD were validated and verified individually by comparison with sample hand calculations. Additionally, the subroutines were again validated and verified as they were added to the parent program. It was this validation and verification procedure which first indicated there were possible problems with Vannman's published B value formula which supported the generation of the best linear unbiased estimates for \( c=1 \). Chapter 3 identifies the published version of the B value formula and the correction required.
Steps in BLU vs MD Estimate Comparison

1. Generate a sample set of \( n \) random variates from a Pareto distribution with location and scale equal to 1 and shape equal to \( c \).

2. Order sample from smallest to largest.

3. Calculate BLU estimates for location and scale based on sample size \( n \).

4. Calculate the Kolmogorov minimum distance estimates of location and scale based on the sample.

5. Calculate the Cramer-von Mises minimum distance estimates of location and scale based on the sample.

6. Calculate the Anderson-Darling minimum distance estimates of location and scale based on the sample.

7. Find the error from the true value of 1 for each estimate and square this error. Save a running sum of the squared error terms for each estimate.

8. Repeat steps 1-7 1000 times for a given \( n \).

9. Divide all eight squared error totals by 1000 to give the MSE's.

10. Output the 14 MSE's for the given \( n \) and \( c \) values.

11. Repeat steps 1-10 using a different sample size, \( n \), but the same \( c \), until all values of \( n \) have been used.

12. Repeat steps 1-11 using a new shape value, \( c \), until all values of \( c \) have been used.

Figure 1. Pseudocode for Program BLUMD
V. Results, Analysis and Conclusions

Figure 1 in Chapter IV described the pseudocode of the computer program used to generate each Pareto random variate sample set, calculate the best linear unbiased and the six new minimum distance estimates for each parameter based on each sample set, and finally determine the mean square error for each estimate. This chapter presents the results of the computer program runs along with an analysis of these results. Appendix A contains the results of the study in table format, where a separate table of Mean Square Error values is presented for each unique shape parameter--sample size combination investigated. Since there were 20 possible combinations of shapes and sample sizes, Appendix A contains 20 separate tables. Finally, this chapter presents the conclusions drawn from the analysis of these results.

RESULTS

Appendix A contains the tables of mean square errors (MSEs) for each estimation technique used in the research effort, given a particular shape parameter and sample size. Since MSE is the evaluation tool used to determine which estimation technique was best, these tables were used to make the estimator comparisons. The estimator with the smallest MSE value is considered the best estimator of those investigated.
TABLE II
Mean square error for $c = 1$ and $n = 9$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>2.1462625E-02</td>
</tr>
<tr>
<td>ADM01</td>
<td>2.3038462E-02</td>
</tr>
<tr>
<td>CMD1</td>
<td>2.4678135E-02</td>
</tr>
<tr>
<td>KSMD1</td>
<td>4.0461082E-02</td>
</tr>
<tr>
<td>CMD2</td>
<td>4.2301375E-02</td>
</tr>
<tr>
<td>KSMD2</td>
<td>4.3836664E-02</td>
</tr>
<tr>
<td>ADM2</td>
<td>6.1653644E-02</td>
</tr>
</tbody>
</table>

Figure 2. Sample Table of Mean Square Errors

Figure 2 shows a sample of the table format. Each table contains two sections. The left section contains the MSEs based upon estimation of the location parameter while the right section contains the MSEs based upon estimation of the scale parameter. This format permits an easy comparison of the BLUE MSE value for each parameter with the MSE value of each of the minimum distance estimation techniques used. The smallest MSE value in each table section then reflects the best estimation technique to use for that parameter, under the stated shape parameter and sample size conditions. To further simplify the reading of the tables, the estimation techniques are ordered in each table from smallest MSE value to largest. Therefore, the technique which generated the smallest MSE result is listed first in each column and is also the
best estimation technique to use for estimating that parameter under the specified conditions.

Each section of the table contains a list of the estimation techniques that were applied to the 1000 sample sets of ordered Pareto random variates. BLUE refers to the best linear unbiased estimation technique. Each of the other techniques was compared against this technique to determine which technique provided the better estimate. ADMD1 refers to the Anderson-Darling minimum distance estimation technique. Additionally, the 1 implies that only one parameter was permitted to vary while the other parameter was held constant (equal to the BLU estimate). For example, ADMD1 under the location parameter section of the table implies that the location parameter was varied while the scale parameter was held equal to the BLU estimate for that sample size and shape parameter. CVMD1 and KSMD1 refer to the Cramer-von Mises and the Kolmogorov minimum distance estimation techniques respectively. Again, the 1 implies that only one parameter was allowed to vary in finding the minimum distance value, while the other parameter was held equal to the BLU estimate. ADMD2 again refers to the use of the Anderson-Darling distance measure in the minimization process. However, in this case, both the location and scale parameters were permitted to vary simultaneously in determining the minimum distance measure. The 2 in the notation indicates that two parameters were allowed to vary during the minimization process. CVMD2 and KSMD2 refer to the use of the Cramer-von Mises and Kolmogorov distance measures respectively. Again, the 2 in the notation implies that two parameters (i.e., location and scale) were permitted to vary during the
minimization routine. Therefore, in this report, ADMD1, CVMD1 and KSMD1 are called single parameter minimum distance estimation techniques while ADMD2, CVMD2 and KSMD2 are called double parameter minimum distance estimation techniques.

ANALYSIS

Regarding the location parameter, the BLU estimator provided the smallest MSE values in all cases except the case where the shape parameter equalled 1 (c = 1) and the sample size equalled 6 (n = 6). In this case, the single parameter Anderson-Darling minimum distance estimator (ADMD1) provided the smallest MSE. Based on this analysis, the results showed that, overall, the best linear unbiased estimator performed better than any of the minimum distance estimators evaluated.

The results of the research regarding estimation of the scale parameter was even more pronounced. Regardless of the shape parameter (c = 1, 2, 3 or 4) or sample size (n = 6, 9, 12, 15, or 18) used in this study, the BLUE provided the smallest MSE in every case and is therefore ranked as the best of the estimation techniques investigated. None of the minimum distance estimation techniques provided better MSE values in any instance. Therefore, investigators should feel comfortable using the BLUE as an instrument of estimation when the underlying population distribution is the Pareto.

Additionally, some observations were made regarding the minimum distance estimation techniques that were applied in this study and how they performed against each other. Performance of the minimum distance estimators on both location and scale were addressed.
For the location parameter, the single parameter Anderson-Darling
minimum distance estimator (ADMD1) provided the smallest MSE values in
every case among the minimum distance estimators tested. ADMD1 was
therefore considered the best minimum distance estimator of location
among those investigated.

One concern this researcher had regarding the minimum distance
estimation technique was whether to let both the location and scale
parameters vary (double parameter estimator) to achieve the minimum
distance measure or to permit only one of the parameters to vary (single
parameter estimator) while holding the other as a constant, equal to the
BLUE for that parameter. For the location parameter, the results show
that the single parameter minimum distance estimator out-performed its
double parameter counterpart in every case except one. When \( c=1 \) and
\( n=6 \), KSMD2 provided a smaller MSE than did KSMD1. In all other cases,
however, the single parameter minimum distance estimator provided better
results. Therefore, the single parameter estimation technique performed
better than its double parameter counterpart when the Kolmogorov,
Cramer-von Mises, or Anderson-Darling distance measure was minimized for
location parameter estimation of the Pareto.

For the scale parameter, the inferences drawn required a bit more
scrutiny as there was no single best minimum distance estimator. There
was a shift in performance when a shape parameter \( c > 1 \) was specified.
For \( c = 1 \), CVMD1 was the best overall estimator, since it provided the
smallest MSE in four of the five cases investigated. The exception
occurred again in the case \( c=1 \) and \( n=6 \), where the KSMD1 estimator
gave the smallest MSE; however, CVMD1 did provide the next smallest MSE.
Therefore, CVMD1 was selected as the best minimum distance estimator for scale when the shape was specified as \( c = 1 \). In the other 15 cases investigated, KSMD2 provided the smallest MSE values in 12 instances. The three exceptions were: \( c = 2 \) and \( n = 6 \) where KSMD1 was best, \( c = 2 \) and \( n = 15 \) where CVMD1 was best, and \( c = 3 \) and \( n = 15 \) where ADMD1 was best. Overall, CVMD1 performed best for \( c = 1 \) and KSMD2 performed best for \( c = 2, 3 \) or 4 among the minimum distance estimators for scale investigated.

Regarding use of the single parameter versus the double parameter minimum distance estimation technique for scale, no clear rule can be stated, although there was a definite trend shown in the results. For \( c = 1 \), the single parameter technique clearly dominated since in all but one case, the single parameter estimator provided smaller MSE values than the corresponding double parameter estimator. The only exception was for \( c = 1 \) and \( n = 12 \) where KSMD2 performed better than KSMD1 (perhaps an indication of the improved performance this estimator would show for larger \( c \) values). However, as the shape parameter value increased from 1 to 4, the performances of the double parameter techniques improved. In fact for \( c = 4 \), the double parameter estimation techniques performed better than their single parameter counterparts in all but one case: for \( c = 4 \) and \( n = 18 \), ADMD1 out-performed ADMD2. Therefore, for \( c = 1 \), the single parameter minimum distance estimators performed better overall than their double parameter counterparts. For \( c = 4 \), the reverse was true. For shape values of 2 and 3, the performance was mixed, but the trend toward improved double parameter performance with the increasing value of shape was still evident.
CONCLUSION

The observations made regarding the minimum distance estimators, although lengthy, should not overshadow the primary conclusion drawn from this research. The best linear unbiased estimators provided the best estimates of both location and scale when compared with any of the minimum distance estimators based upon the mean square error criteria.
VI. Summary and Recommendations

This chapter presents a summary of the research effort, restating the objective of the study, the methodology used, and the major conclusions drawn from the experimental results. Further, three recommendations for further study in this area are presented.

SUMMARY

The purpose of this research was to compare the minimum distance estimation technique with the best linear unbiased estimation technique to determine which estimator provided more accurate estimates of the underlying location and scale parameter values for a given three parameter Pareto distribution with specified shape parameter. The Kolmogorov, Cramer-von Mises, and Anderson-Darling distance measures were used to develop the minimum distance estimators. For each of these distance measures, two minimum distance estimators were developed. The first minimum distance estimator varied only a single parameter value to achieve the distance measure minimization. This estimator was called the single parameter minimum distance estimator. The second minimum distance estimator allowed both the location and scale parameters to vary while achieving the minimum distance measure. These estimators were called the double parameter minimum distance estimators. These minimum distance estimators were compared against the best linear unbiased estimators which had been previously developed by Kulldorff and Vännman for shape greater than 2, and by Vännman for shape equal to or less than 2. Manual derivation of the B values formula for the special
case, \( c=1 \), revealed an error in the published version of the formula, as explained in Chapter III. of this report.

A Monte Carlo methodology was used to generate the estimates for each of the estimation techniques investigated. A sample of Pareto random variates was generated from a completely specified three parameter Pareto distribution with location and scale equal to one and the shape parameter iteratively specified as one, two, three, or four. The estimates of location and scale were then generated based on each of the estimation techniques. This process was repeated 1000 times for each combination of shape parameter (\( c = 1, 2, 3, \) or 4) and Pareto random variate sample size (\( n = 6, 9, 12, 15, \) or 18). This Monte Carlo process resulted in 1000 estimates of both location and scale for each estimation technique used.

The criteria for determining which estimation technique performed best was based on the resulting mean square error calculation for each group of 1000 estimates. The estimation technique which yielded the smallest mean square error was selected as the best performing estimator.

The results of this research clearly indicated that the best linear unbiased estimator provided smaller mean square error terms than any of the minimum distance estimation techniques investigated. Therefore, the best linear unbiased estimation technique was ranked as the best estimation technique among those tested.

Regarding the minimum distance estimators, a comparison of the single versus double parameter techniques was made. For estimation of the location parameter, the single parameter estimation technique
performed better than the double parameter estimation technique. For the scale parameter estimation, the conclusion was not as clear. A trend was identified as the value of the specified shape parameter increased from 1 to 4. For $c = 1$, the single parameter estimators performed better; however, as the shape parameter increased, the performance of the double parameter estimators improved until at $c = 4$, the double parameter estimators performed better than their single parameter counterparts.

RECOMMENDATIONS

Three recommended areas for further study in this research area are now offered. First, a study similar to this one can be performed, again based upon a specified three parameter Pareto distribution, but using minimum distance estimators based on different distance measures. Examples of such distance measures include the Kuiper distance and the Watson distance referenced by M. A. Stephens (35:731). Second, a study involving a comparison of a set of minimum distance estimators against the best linear unbiased estimators based on the more commonly used two parameter form of the Pareto distribution could prove fruitful. Third, a researcher could perform a comparison study involving the maximum likelihood estimator and a set of minimum distance estimators, again based upon the common two parameter form of the Pareto distribution function. Any of these areas would provide fertile ground for the investigative statistical researcher.
### Appendix A

#### Tables of Mean Square Errors

The following notation is used in this appendix:

<table>
<thead>
<tr>
<th>Term</th>
<th>Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best Linear Unbiased Estimator</td>
<td>BLUE</td>
</tr>
<tr>
<td>Anderson-Darling Minimum Distance Estimator</td>
<td>ADMD1</td>
</tr>
<tr>
<td>(Only one varying parameter)</td>
<td></td>
</tr>
<tr>
<td>Cramer-von Mises Minimum Distance Estimator</td>
<td>CVMD1</td>
</tr>
<tr>
<td>(Only one varying parameter)</td>
<td></td>
</tr>
<tr>
<td>Kolmogorov Minimum Distance Estimator</td>
<td>KSMD1</td>
</tr>
<tr>
<td>(Only one varying parameter)</td>
<td></td>
</tr>
<tr>
<td>Anderson-Darling Minimum Distance Estimator</td>
<td>ADMD2</td>
</tr>
<tr>
<td>(Two varying parameters)</td>
<td></td>
</tr>
<tr>
<td>Cramer-von Mises Minimum Distance Estimator</td>
<td>CVMD2</td>
</tr>
<tr>
<td>(Two varying parameters)</td>
<td></td>
</tr>
<tr>
<td>Kolmogorov Minimum Distance Estimator</td>
<td>KSMD2</td>
</tr>
<tr>
<td>(Two varying parameters)</td>
<td></td>
</tr>
<tr>
<td>Location Parameter</td>
<td>a</td>
</tr>
<tr>
<td>Scale Parameter</td>
<td>b</td>
</tr>
<tr>
<td>Shape Parameter</td>
<td>c</td>
</tr>
<tr>
<td>Sample Size</td>
<td>n</td>
</tr>
<tr>
<td>Mean Square Error</td>
<td>MSE</td>
</tr>
</tbody>
</table>

The Monte Carlo analysis involves 1000 iterations for the generation of each table. The true value of the location parameter is one and the true value of the scale parameter is one for all of the tables.
### TABLE I

Mean square error for $c = 1$ and $n = 6$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
<th>TECHNIQUE</th>
<th>MSE</th>
<th>TECHNIQUE</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADMD1</td>
<td>6.1960317E-02</td>
<td>BLUE</td>
<td>0.9352127</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVMD1</td>
<td>6.4573184E-02</td>
<td>KSMD1</td>
<td>1.460597</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BLUE</td>
<td>6.7886792E-02</td>
<td>CVMD1</td>
<td>1.523946</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVMD2</td>
<td>9.8166950E-02</td>
<td>ADM02</td>
<td>1.589061</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSMD2</td>
<td>1.192953E-01</td>
<td>KSMD2</td>
<td>1.567526</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSMD1</td>
<td>1.2137F4E-01</td>
<td>CVMD2</td>
<td>1.567902</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM02</td>
<td>4.874095E-01</td>
<td>ADM02</td>
<td>3.635443</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### TABLE II

Mean square error for $c = 1$ and $n = 9$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
<th>TECHNIQUE</th>
<th>MSE</th>
<th>TECHNIQUE</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLUE</td>
<td>2.1462625E-02</td>
<td>BLUE</td>
<td>0.4062596</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADMD1</td>
<td>2.3038462E-02</td>
<td>CVMD1</td>
<td>0.5638790</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVMD1</td>
<td>2.4678135E-02</td>
<td>ADM02</td>
<td>0.5760345</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSMD1</td>
<td>4.0461082E-02</td>
<td>KSMD1</td>
<td>0.5824831</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVMD2</td>
<td>4.2301375E-02</td>
<td>KSMD2</td>
<td>0.6407809</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSMD2</td>
<td>4.3836664E-02</td>
<td>CVMD2</td>
<td>0.6685479</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM02</td>
<td>6.1653644E-02</td>
<td>ADM02</td>
<td>0.8771174</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### TABLE III

Mean square error for $c = 1$ and $n = 12$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>TECHNIQUE</td>
</tr>
<tr>
<td>BLUE</td>
<td>1.0628480E-02</td>
</tr>
<tr>
<td>ADMD1</td>
<td>1.1631799E-02</td>
</tr>
<tr>
<td>CVMD1</td>
<td>1.3175875E-02</td>
</tr>
<tr>
<td>CVMD2</td>
<td>2.2465685E-02</td>
</tr>
<tr>
<td>KSMD1</td>
<td>2.3550959E-02</td>
</tr>
<tr>
<td>KSMD2</td>
<td>2.4744846E-02</td>
</tr>
<tr>
<td>ADM2</td>
<td>3.2977652E-02</td>
</tr>
</tbody>
</table>

### TABLE IV

Mean square error for $c = 1$ and $n = 15$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>TECHNIQUE</td>
</tr>
<tr>
<td>BLUE</td>
<td>7.4139438E-03</td>
</tr>
<tr>
<td>ADMD1</td>
<td>9.5694885E-03</td>
</tr>
<tr>
<td>CVMD1</td>
<td>1.1153221E-02</td>
</tr>
<tr>
<td>KSMD1</td>
<td>1.6373332E-02</td>
</tr>
<tr>
<td>KSMD2</td>
<td>2.4338266E-02</td>
</tr>
<tr>
<td>CVMD2</td>
<td>2.5020871E-02</td>
</tr>
<tr>
<td>ADM2</td>
<td>2.9120248E-32</td>
</tr>
</tbody>
</table>
### TABLE V
Mean square error for \( c = 1 \) and \( n = 18 \)

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>4.3005478E-03</td>
</tr>
<tr>
<td>ADM1</td>
<td>5.2362322E-03</td>
</tr>
<tr>
<td>CVMD1</td>
<td>6.4189113E-03</td>
</tr>
<tr>
<td>ADM2</td>
<td>8.6196102E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>1.0911038E-02</td>
</tr>
<tr>
<td>CVMD2</td>
<td>1.1255259E-02</td>
</tr>
<tr>
<td>KSMD2</td>
<td>1.2906388E-02</td>
</tr>
</tbody>
</table>

### TABLE VI
Mean square error for \( c = 2 \) and \( n = 6 \)

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>1.1706045E-02</td>
</tr>
<tr>
<td>ADM1</td>
<td>1.2802768E-02</td>
</tr>
<tr>
<td>CVMD1</td>
<td>1.3657569E-02</td>
</tr>
<tr>
<td>KSMD1</td>
<td>1.5005208E-02</td>
</tr>
<tr>
<td>CVMD2</td>
<td>1.7071828E-02</td>
</tr>
<tr>
<td>KSMD2</td>
<td>2.2394231E-02</td>
</tr>
<tr>
<td>ADM2</td>
<td>2.8150991E-02</td>
</tr>
</tbody>
</table>
### TABLE VII

Mean square error for $c = 2$ and $n = 9$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>4.6055736E-03</td>
</tr>
<tr>
<td>ADM1</td>
<td>5.15687085E-03</td>
</tr>
<tr>
<td>CVMD1</td>
<td>5.9638712E-03</td>
</tr>
<tr>
<td>CVMD2</td>
<td>6.9629652E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>6.963503E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>7.8367852E-03</td>
</tr>
<tr>
<td>ADM2</td>
<td>8.9378590E-03</td>
</tr>
</tbody>
</table>

### TABLE VIII

Mean square error for $c = 2$ and $n = 12$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>1.9467097E-03</td>
</tr>
<tr>
<td>ADM1</td>
<td>2.2874754E-03</td>
</tr>
<tr>
<td>CVMD1</td>
<td>2.9733009E-03</td>
</tr>
<tr>
<td>ADM2</td>
<td>3.0276354E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>3.4914461E-03</td>
</tr>
<tr>
<td>CVMD2</td>
<td>3.7764474E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>4.6899226E-03</td>
</tr>
</tbody>
</table>
TABLE IX
Mean square error for $c = 2$ and $n = 15$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>ESTIMATION</td>
</tr>
<tr>
<td>TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td></td>
<td>TECHNIQUE</td>
</tr>
<tr>
<td></td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>1.2336310E-03</td>
</tr>
<tr>
<td>ADMD1</td>
<td>1.4523189E-03</td>
</tr>
<tr>
<td>CVM1</td>
<td>1.9546540E-03</td>
</tr>
<tr>
<td>ADMD2</td>
<td>2.0549898E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>2.3178468E-03</td>
</tr>
<tr>
<td>CVM2</td>
<td>2.6572768E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>3.2847996E-03</td>
</tr>
</tbody>
</table>

TABLE X
Mean square error for $c = 2$ and $n = 18$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>ESTIMATION</td>
</tr>
<tr>
<td>TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td></td>
<td>TECHNIQUE</td>
</tr>
<tr>
<td></td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>7.7167596E-04</td>
</tr>
<tr>
<td>ADMD1</td>
<td>9.3843893E-04</td>
</tr>
<tr>
<td>ADMD2</td>
<td>1.3170647E-03</td>
</tr>
<tr>
<td>CVM1</td>
<td>1.3989438E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>1.8159606E-03</td>
</tr>
<tr>
<td>CVM2</td>
<td>1.8582373E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>2.3305281E-03</td>
</tr>
</tbody>
</table>
TABLE XI
Mean square error for $c = 3$ and $n = 6$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>4.6162526E-03</td>
</tr>
<tr>
<td>ADMD1</td>
<td>4.7417325E-03</td>
</tr>
<tr>
<td>CVM1</td>
<td>4.996846E-03</td>
</tr>
<tr>
<td>CVM2</td>
<td>5.5936114E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>5.6171883E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>6.154517E-03</td>
</tr>
<tr>
<td>ADMD2</td>
<td>1.7377743E-02</td>
</tr>
</tbody>
</table>

TABLE XII
Mean square error for $c = 3$ and $n = 9$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>2.5504350E-03</td>
</tr>
<tr>
<td>ADMD1</td>
<td>2.932801E-03</td>
</tr>
<tr>
<td>CVM1</td>
<td>3.2298244E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>3.4083289E-03</td>
</tr>
<tr>
<td>CVM2</td>
<td>3.4843122E-03</td>
</tr>
<tr>
<td>ADMD2</td>
<td>3.6204350E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>3.8682341E-03</td>
</tr>
</tbody>
</table>
### TABLE XIII
Mean square error for $c = 3$ and $n = 12$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>1.1032750E-03</td>
</tr>
<tr>
<td>ADM01</td>
<td>1.3244717E-03</td>
</tr>
<tr>
<td>ADM02</td>
<td>1.5484567E-03</td>
</tr>
<tr>
<td>CVMD1</td>
<td>1.6325671E-03</td>
</tr>
<tr>
<td>CVMD2</td>
<td>1.8109155E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>1.8245766E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>2.1554409E-03</td>
</tr>
</tbody>
</table>

### TABLE XIV
Mean square error for $c = 3$ and $n = 15$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>6.1996793E-04</td>
</tr>
<tr>
<td>ADM01</td>
<td>8.2581321E-04</td>
</tr>
<tr>
<td>ADM02</td>
<td>1.0506152E-03</td>
</tr>
<tr>
<td>CVMD1</td>
<td>1.0002834E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>1.2737850E-03</td>
</tr>
<tr>
<td>CVMD2</td>
<td>1.2966762E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>1.5728395E-03</td>
</tr>
</tbody>
</table>
### TABLE XV

Mean square error for $c = 3$ and $n = 18$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>MSE</td>
</tr>
<tr>
<td>TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>3.5388739E-04</td>
</tr>
<tr>
<td>ADM01</td>
<td>4.7279926E-04</td>
</tr>
<tr>
<td>ADM02</td>
<td>5.8612920E-04</td>
</tr>
<tr>
<td>CVMD1</td>
<td>6.7394995E-04</td>
</tr>
<tr>
<td>KSMD1</td>
<td>7.9139235E-04</td>
</tr>
<tr>
<td>CVMD2</td>
<td>8.1001956E-04</td>
</tr>
<tr>
<td>KSMD2</td>
<td>1.0424773E-03</td>
</tr>
</tbody>
</table>

### TABLE XVI

Mean square error for $c = 4$ and $n = 6$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>MSE</td>
</tr>
<tr>
<td>TECHNIQUE</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>2.2004284E-03</td>
</tr>
<tr>
<td>ADM01</td>
<td>2.3699524E-03</td>
</tr>
<tr>
<td>CVMD1</td>
<td>2.5871816E-03</td>
</tr>
<tr>
<td>CVMD2</td>
<td>2.6604387E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>2.7944313E-03</td>
</tr>
<tr>
<td>ADM02</td>
<td>3.0173361E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>3.1544713E-03</td>
</tr>
</tbody>
</table>
## TABLE XVII
Mean square error for $c = 4$ and $n = 9$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>9.8934094E-04</td>
</tr>
<tr>
<td>ADM1</td>
<td>1.1494281E-03</td>
</tr>
<tr>
<td>CVM1</td>
<td>1.3984025E-03</td>
</tr>
<tr>
<td>ADM2</td>
<td>1.4030078E-03</td>
</tr>
<tr>
<td>CVM2</td>
<td>1.4857050E-03</td>
</tr>
<tr>
<td>KSMD1</td>
<td>1.5590133E-03</td>
</tr>
<tr>
<td>KSMD2</td>
<td>1.7792400E-03</td>
</tr>
</tbody>
</table>

## TABLE XVIII
Mean square error for $c = 4$ and $n = 12$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ESTIMATION</td>
<td>MSE</td>
</tr>
<tr>
<td>BLUE</td>
<td>4.3847732E-04</td>
</tr>
<tr>
<td>ADM1</td>
<td>5.3575670E-04</td>
</tr>
<tr>
<td>ADM2</td>
<td>6.4325968E-04</td>
</tr>
<tr>
<td>CVM1</td>
<td>7.0931221E-04</td>
</tr>
<tr>
<td>CVM2</td>
<td>8.0394866E-04</td>
</tr>
<tr>
<td>KSMD1</td>
<td>8.4389572E-04</td>
</tr>
<tr>
<td>KSMD2</td>
<td>9.7926693E-04</td>
</tr>
</tbody>
</table>
### TABLE XIX
Mean square error for $c = 4$ and $n = 15$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
<th>TECHNIQUE</th>
<th>MSE</th>
<th>TECHNIQUE</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLUE</td>
<td>2.8805208E-04</td>
<td>BLUE</td>
<td>0.1210762</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM1</td>
<td>3.7486354E-04</td>
<td>KSMD2</td>
<td>0.1318542</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM2</td>
<td>4.3274325E-04</td>
<td>CVMD2</td>
<td>0.1393966</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVMD1</td>
<td>5.3938437E-04</td>
<td>ADM1</td>
<td>0.1417205</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVMD2</td>
<td>6.0927385E-04</td>
<td>ADM0</td>
<td>0.1437359</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSMD1</td>
<td>6.2738883E-04</td>
<td>CVMD1</td>
<td>0.1477181</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSMD2</td>
<td>7.4160466E-04</td>
<td>KSMD1</td>
<td>0.1574258</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### TABLE XX
Mean square error for $c = 4$ and $n = 18$

<table>
<thead>
<tr>
<th>LOCATION (a)</th>
<th>SCALE (b)</th>
<th>TECHNIQUE</th>
<th>MSE</th>
<th>TECHNIQUE</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLUE</td>
<td>1.9949843E-04</td>
<td>BLUE</td>
<td>9.3522320E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM1</td>
<td>2.6865197E-04</td>
<td>KSMD2</td>
<td>1.0103040E-01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADM2</td>
<td>2.9910799E-04</td>
<td>ADM1</td>
<td>1.0706935E-01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVMD1</td>
<td>4.2770977E-04</td>
<td>ADM2</td>
<td>1.0771950E-01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CVMD2</td>
<td>4.5176961E-04</td>
<td>CVMD2</td>
<td>1.0861408E-01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSMD1</td>
<td>5.0622342E-04</td>
<td>CVMD1</td>
<td>1.1239370E-01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KSMD2</td>
<td>5.9582230E-04</td>
<td>KSMD1</td>
<td>1.1357192E-01</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Appendix B

Computer Program for Estimator Comparison

The following FORTRAN computer program, BLUMD.FOR, was written to perform the Monte Carlo analysis and to generate the mean square errors for each estimation technique investigated. Program documentation is included within the program as comment statements to inform the reader of the purpose of each statement or group of statements. Additionally, each subroutine is prefaced by extensive documentation to inform the reader of the purpose of the subroutine, all of the variables used in the subroutine, the input variables required, the output variables generated, and the major computations performed within the subroutine to obtain the desired outputs.
BLUMD (BLU/MINIMUM DISTANCE) MAIN PROGRAM

Purpose: BLUMD calculates the best linear unbiased estimates in addition to the six minimum distance estimates (based on the Kolmogorov, the Cramer-von Mises, and the Anderson-Darling distances) for both the location and scale parameters of the three parameter Pareto distribution, where the shape parameter is varied between the integer values 1, 2, 3 and 4. Sample sizes of 6, 9, 12, 15, and 18 are used. Pareto variates are generated for each combination of shape parameter and sample size. Finally, BLUMD calculates the mean square error for each estimate type to compare which estimation technique performs best.

Variables: 
- n = sample size
- c = shape parameter
- nn = sample size symbol (varies from 1-5 representing each permissible sample size)
- kkk = dummy variable used to convert nn to n by using the formula: kkk = 3 + (nn*3)
- dseed = double precision seed for the Pareto variates
- x = array of Pareto variates
- B = array of B values used to calculate the blues for shape greater than 2
- BB = array of BB values used to calculate the blues for shape less than or equal to 2
- D = constant used to calculate the blues for shape greater than 2
- Anc = constant used to calculate the blues for shape less than or equal to 2
- Bnc = constant used to calculate the blues for shape less than or equal to 2
- ablu = blu estimate of location, a
- bblu = blu estimate of scale, b
- aKS = Kolmogorov minimum distance estimate for a
- bKS = Kolmogorov minimum distance estimate for b
- aCVM = Cramer-von Mises min distance estimate for a
- bCVM = Cramer-von Mises min distance estimate for b
- aAD = Anderson-Darling min distance estimate for a while holding b = bblu as constant
- bAD = Anderson-Darling min distance estimate for b while holding a = ablu as constant
- a2AD = Anderson-Darling min distance estimate for a
- b2AD = Anderson-Darling min distance estimate for b
- aICV = Cramer-von Mises min distance estimate for a while holding b = bblu as constant
- bICV = Cramer-von Mises min distance estimate for b while holding a = ablu as constant
- a1KS = Kolmogorov minimum distance estimate for a while holding b = bblu as constant
- b1KS = Kolmogorov minimum distance estimate for b while holding a = ablu as constant
c sse = array of sum of squared errors for each
c estimation technique where the true value of
a is 1 and the true value of b is 1.
c mse = array of mean square errors for each
estimation technique used
c count = array of counters used to count the number of
valid estimate values found
c anda = array of calculated A-D distance measures
when estimating location alone
c esta = array of location estimates used to minimize
the A-D distance
c icnt = counter for the number of location estimates
used to minimize the A-D distance
c anda = array of calculated A-D distance measures
when estimating scale alone
c estb = array of scale estimates used to minimize
the A-D distance
c icntb = counter for the number of scale estimates
used to minimize the A-D distance
c andab = array of calculated A-D distance measures
when estimating a and b simultaneously
c esta = array of location estimates used to minimize
the A-D distance
c estb = array of scale estimates used to minimize
the A-D distance
c icntab = counter for the number of location and scale
estimates used to minimize the A-D distance

C Inputs: dseed = double precision seed for Pareto variate
c generation
c c = shape parameter
c n = sample size

C Outputs: mse = array of mean square errors for each
c estimation technique for each parameter under
investigation (location and scale)

C Calculate: mse = sse/number of trials

*** Variable Declarations
common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,
1 aCVM,bCVM,aAD,bAD,nn,count,a2AD,b2AD,anda,esta,icnt
1 ,andb,estb,icntb,andab,estaa,estbb,icntab
1 ,aICV,bICV,aIKS,bIKS
integer n,nn,count(4,5,14),c,kkk,icnt,icntb,icntab
real x(18),ablu,bblu,D(18),D,Anc,Bnc,BB(18),aKS,bKS,
1 aCVM,bCVM,aAD,bAD,sse(4,5,14),mse(4,5,14),a2AD,b2AD
1 ,anda(500),andb(500),andab(500),esta(500),estb(500),
1 estaa(500),estbb(500),aICV,bICV,aIKS,bIKS
double precision dseed
call uerset(0,levold)
dseed = 43846219217.d00
print*,dseed = ',dseed

72
do 90 i=1,4
  c = 1
  nn = 0
  do 80 j=6,18,3
    n = j
    nn = nn + 1
    do 40 jj=1,14
      sse(c,nn,jj) = 0
  40 continue
  do 70 it = 1,1000
    if ((it.eq.200) .or. (it.eq.400) .or. (it.eq.600) .or.
      (it.eq.800) .or. (it.eq.1000)) then
      print*, 'c=',c,' n=',n,' iteration=',it
    end if
    call PARVAR
    if (c .gt. 2) then
      call BCGT2
      call 9LCGT2
      go to 45
    end if
    call BCLE2
    call BLCLE2
  45 if (ablu .eq. 0 .and. bblu .eq. 0) then
    go to 70
  end if
  call KSMD
  call CVMMD
  call ADMD
  call AD2MD
  call CVAMD
  call CVMMD
  call KSAMD
  call KSBMD

  c    *** Calculate the Sum of Squared Errors
  58  sse(c,nn,1) = sse(c,nn,1) + (ablu - 1)**2
  sse(c,nn,2) = sse(c,nn,2) + (bblu - 1)**2
  sse(c,nn,3) = sse(c,nn,3) + (aKS - 1)**2
  sse(c,nn,4) = sse(c,nn,4) + (bKS - 1)**2
  sse(c,nn,5) = sse(c,nn,5) + (aCVM - 1)**2
  sse(c,nn,6) = sse(c,nn,6) + (bCVM - 1)**2
  sse(c,nn,7) = sse(c,nn,7) + (aAD - 1)**2
  sse(c,nn,8) = sse(c,nn,8) + (bAD - 1)**2
  sse(c,nn,9) = sse(c,nn,9) + (a2AD - 1)**2
  sse(c,nn,10) = sse(c,nn,10) + (b2AD - 1)**2
  sse(c,nn,11) = sse(c,nn,11) + (a1CV - 1)**2
  sse(c,nn,12) = sse(c,nn,12) + (b1CV - 1)**2
  sse(c,nn,13) = sse(c,nn,13) + (a1KS - 1)**2
  sse(c,nn,14) = sse(c,nn,14) + (b1KS - 1)**2
  if (it .eq. 1000) then
    continue
  end if

  c    *** Calculate the mean square error for each estimate type
  60  do 11 = 1,14
    kkk = 3 + (nn*3)
if (count(c,nn,11) .eq. 0) then
  print*, 'count=0 for c=',c,' n=',kkk,' est=',ll
  go to 60
end if
mse(c,nn,11) = sse(c,nn,11)/count(c,nn,11)
print*, 'mse=',mse(c,nn,11), ' c=',c,' n=',kkk,' est=',ll
print*, 'count=',count(c,nn,11)
60 continue
end if
90 continue
continue
90 continue
continue
end

Subroutine PARVAR

C------------------------------------------------------------------
c Purpose: For a specified sample size, n, PARVAR
generates n random variates from a Pareto
distribution with location and scale parameters
set equal to one and the shape parameter, c,
set to either 1, 2, 3 or 4.
C--------------------------------------------------------------------------------
c Formula: x = (1/r)**(1/c)
c---------------------------------
C Variables.
c      r = random number
c      z = shape parameter
c      x = array of Pareto variates
c      n = sample size
c      dseed = random number seed
c-------------------
C Inputs: dseed = random number seed
c      c = shape parameter
c      n = sample size
c-------------------
C Outputs: x = array of Pareto random variates
c-------------------
C Calculate:
c      x(j) = (1/r(j))** (1/c)
c-------------------

*** Variable Declarations
real r(18),x(18),ablu,bblu,B(18) *A,anc,bnc,bb(18),aKS,bKS,
aCVM,bCVM,aAD,aAD,a2AD,b2AD
integer n,c,nn,count(4,5,14)
common n,x,c,ablu,bblu,dseed,B,anc,bnc,bb,aKS,bKS,
aCVM,bCVM,aAD,aAD,nn,count,a2AD,b2AD
double precision dseed
do 10 j=1,n
c *** Call IMSL random number generator subroutine ggubs
call ggubs(dseed,n,r)
c *** Use the inverse transform technique for Pareto variates
x(j)=(1/r(j))** (1/real(c))
10 continue
Call VSRTA to sort the variates in ascending order

call vsrta(x,n)
return
end

Subroutine BCLEZ

Purpose: For a given sample size, n, and a specified shape
(c=1 or c=2); BCLEZ calculates the B values used
to find the blue estimates of location and scale.
In addition, it calculates the constants Anc and Bnc
for the given shape and sample size.

Variables:  
c = shape parameter
n = sample size
BB = array of B values (k in number)
k = number of order statistics used; k=n-[2/c]
nc = product of n and c
Anc = constant in the formula for the blue for scale
Bnc = constant in the formula for the blue for scale

Inputs:  
c = shape parameter
n = sample size

Outputs:  
BB = array of B values
Anc = constant in the blue for the scale parameter
Bnc = constant in the blue for the scale parameter

Calculate:
Anc = (c+1)(c+2)(nc-1) / (nc-2)(nc-c-2)
Bnc = (nc-2) / (c+2)

For c=1:  
B(1) = (1 - 1/n) [ 1 - 1/(n-1) ]

For c=2:  
B(1) = 1 - 1/n

Variable Declarations:
real nc,x(18),ablu,bblu,B(18),D,Anc,Bnc,BB(18),aKS,bKS,  
aCVM,bCVM,aAD,bAD,aZAD,bZAD
integer n,k,c,nn,count(4,5,14)
double precision dseed
common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,bb(18),aKS,bKS,  
aCVM,bCVM,aAD,bAD,aZAD,bZAD
k=n-(2/c)

Calculate the B values when c=1
if (c.eq.1) then
  do 10 j=1,k
   BB(j)=(1-j/real(n))**1-j/(real(n)-1))
 10 continue
end if

Calculate the B values when c=2 (i.e., c.ne.1)
do 20 m=1,k
    BB(m)=1-m/real(n)
20  continue
36  nc = n * c

*** Calculate the constants Anc and Bnc
Anc= ((c+1)*(c+2)*(nc-1)) / ((nc-Z)*(nc-c-Z))
Bnc= (nc-Z) / (c+2)
return
end

Subroutine BLCLLEZ

Purpose: For a given sample size, n, and shape (c=1 or c=2),
BLCLLEZ calculates the best linear unbiased estimates
of location and scale for a sample of ordered Pareto
variates.

Variables: x = array of ordered Pareto variates
c = shape parameter
n = sample size
BB = array of B values used to calculate the blues
nc = product of n and c
k = number of order statistics used; k = n - [Z/c]
Anc = constant used to calculate the blue for scale
Bnc = constant used to calculate the blue for scale
Bxsum = sum of [B(i) * x(i)] terms for i = 1,2,...,k
ablu = blu estimate of the location parameter, a
bblu = blu estimate of the scale parameter, b
count = array of counters used to count the number of
valid estimate values found

Inputs: x = array of ordered Pareto variates
c = shape parameter
n = sample size
BB = array of B values used to calculate the blues
Anc = constant used to calculate the blue for scale
Bnc = constant used to calculate the blue for scale

Outputs: ablu = blu estimate of the location parameter, a
bblu = blu estimate of the scale parameter, b

Calculate:
b = (Anc) [ B(1)x(1) + B(2)x(2) + ... + B(k)x(k)]
   - (Bnc)x(1) ]

a = x(1) - [ b/(nc-1) ]

*** Variable Declarations:
real x(:,.),ablu,bblu,Bxsum,nc,B(18),Anc,Bnc,BB(18),
aKS,bKS,aCVM,bCVM,aAD,bAD,a2AD,b2AD
integer n,k,c,nn,count(4,5,14)
double precision dseed
common n,x,c,ablu,bblu,seed,B,D,Anc,Bnc,BB,aKS,bKS,
```fortran

1   aLVM, bCVM, aAD, bAD, nn, count, aZAD, bZAD
   Bxsum = 0
   k = n-(Z/c)
   *** Sum the products of B(i) and x(i) for i = 1, 2, ..., k
   do 10 j=1, k
      Bxsum = Bxsum + BB(j) * x(j)
   10 continue
   nc = n*c
   *** Calculate the blue for scale, then for location
   bblu = Anc*(Bxsum - Bnc*x(1))
   ablu = x(1) - bblu/(nc-1)
   *** Increment counter for valid blues
   if (bblu .gt. 0) then
      count (c, nn, 1) = count (c, nn, 1) + 1
      count (c, nn, 2) = count (c, nn, 2) + 1
   else
      print *, 'bblu=', bblu, 'ablu=', ablu, ' negativity'
      ablu = 0
      bblu = 0
   end if
   return
end

Subroutine BCGTZ

c--------------------------------------------------
c Purpose: For a given sample size, n, and a specified shape, c c0Z; BCGTZ calculates the B and D values used to c find the blu estimates of location and scale.
c--------------------------------------------------
c Variables: c = shape parameter
n = sample size
B = array of B values (n in size)
D = D value
bsum = sum of B values for i=1 . . . (n-1)
c Inputs: c = shape parameter
n = sample size
c Outputs: B = array of B values
D = D value
c Calculate:
B(i) = [1 - Z/c(n-1+i)] * B(i-1)
D = (c+1)[B(1) + B(2) + ... + B(n-1)] + (c-1)B(n)
c--------------------------------------------------

*** Variable Declarations:
real bsum, x(18), ablu, bblu, B(18), D, Anc, Bnc, BB(18), aKS, bKS,
aCVM, bCVM, aAD, bAD, aZAD, bZAD
integer n, c, nn, count (4, 5, 14)
double precision dseed
common n, x, c, ablu, bblu, dseed, B, D, Anc, Bnc, BB, aKS, bKS,
aCVM, bCVM, aAD, bAD, nn, count, aZAD, bZAD
```
c *** Calculate the first B value
B(1)=(1-(2/(c*real(n))))

c *** Calculate the second thru the nth B values
do 10 j=Z,n
   B(j)=B(j-1)*(1-(2/(real(c)*(n-j+1))))
10 continue
bsum=0

c *** Sum the 'first' to the 'nth minus one' B values
do 20 k=1,(n-1)
   bsum=bsum+B(k)
20 continue

c *** Calculate the D value
D = (c+1) * bsum + (c-1) * B(n)
return
end

Subroutine BLCGTZ

Purpose: For a given sample size, n, and a specified shape, c\( \gamma \), BLCGTZ calculates the best linear unbiased c estimates of location and scale.

Variables: 
- \( x \) = array of ordered Pareto variates
- \( \gamma \) = shape parameter
- \( n \) = sample size
- \( B \) = array of B values used to calculate the blues
- \( D \) = D value used to calculate the blues
- \( Y \) = Y value used to calculate the blues
- \( a \) = blu for location parameter, a
- \( b \) = blu for scale parameter, b
- \( B_xsum \) = sum of \( B(i) \times x(i) \) terms for \( i = 1, 2, ..., (n-1) \)
- \( nc \) = product of \( n \) and \( c \)
- \( count \) = array of counters used to count the number of valid estimate values found

Inputs: 
- \( x \) = array of ordered Pareto variates
- \( \gamma \) = shape parameter
- \( n \) = sample size
- \( B \) = array of B values used to calculate blues
- \( D \) = D value used to calculate the blues

Outputs: 
- \( ablu \) = blu estimate for location, a
- \( bblu \) = blu estimate for scale, b

Calculate:
\[
Y = \frac{(c+1)[ B(1)x(1) + B(2)x(2) + \ldots + B(n-1)x(n-1)]}{(nc-1)\{ x(1) - Y/(nc-1)(nc-2) - Dncl \}}
\]
\[
a = x(1) - Y/[(nc-1)(nc-2) - Dncl]
\]
\[
b = (nc-1)\{ x(1) - a \}
\]

*** Variable Declarations:
real x(18), abl, bblu, Bxsum, nc, Y, B(18), D, Anc, Bnc, BB(18)
integer n, c, nn, count(4, 5, 14)
double precision dseed
common n, x, c, abl, bblu, dseed, B, D, Anc, Bnc, BB, aKS, bKS,
aCVM, bCVM, aAD, bAD, aZAD, bZAD

Bxsum = 0

*** Sum the products of the B(i) and x(i) values to i = n-1
do 10 j = 1, (n-1)
   Bxsum = Bxsum + B(j)*x(j)
10 continue

*** Calculate the Y value
   Y = (c + 1)*Bxsum + (c - 1)*B(n)*x(n) - D*x(1)
   nc = n*c

*** Calculate the blu estimates for location and scale
   ablu = x(1) - Y / ((nc - 1)*(nc - 2) - (nc*D))
   bblu = (nc - 1)*(x(1) - ablu)

*** Increment counters for valid blues
   if (bblu .gt. 0) then
      count(c, nn, 1) = count(c, nn, 1) + 1
      count(c, nn, 2) = count(c, nn, 2) + 1
   else
      print*, 'ablu=', ablu, ', bblu=', bblu, ', negativity'
      ablu = 0
      bblu = 0
   end if
end if
return
end

Subroutine KSMO

c----------

 Purpose: KSMO generates the minimum distance estimates of
 location and scale based upon minimizing the
 Kolmogorov distance measure defined in subroutine
 KDIS. This routine uses the blu estimates as the
 starting points for the estimate modifications.

c----------

 Variables: NPAR = number of parameters altered by minimizing
 the Kolmogorov distance
 NSIG = number of significant digits for convergence
 MAXFN = maximum number of function evaluations
 IOPT = options selector (see IMSL manual on ZXMIN)
 H, G, W = vectors defined in IMSL manual on ZXMIN
 IER = error parameter (see IMSL manual on ZXMIN)
 F = value of Kolmogorov distance at the final
 parameter estimates
 kse = Kolmogorov derived minimum distance estimates
 aKS = Kolmogorov minimum distance location estimate
 bKS = Kolmogorov minimum distance scale estimate
 abl = blu estimate of location
 bblu = blu estimate of scale
 count = array of counters used to count the number of
 valid estimate values found

----------
c Inputs:  NPAR = number of parameters altered while minimizing
NSIG = number of significant digits required
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
kse = initial estimates for the minimization process
ablu = blu estimate of location
bblu = blu estimate of scale

c Outputs:  F = minimum value of the function being minimized
kse = revised estimate values
aKS = revised MD estimate of location \{aKS = kse(1)\}
bKS = revised MD estimate of scale \{bKS = kse(2)\}
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)

c Calculate: no calculations performed in this subroutine

*** Variables Declaration:
common n, x, c, ablu, bblu, dseed, B, D, Anc, Bnc, BB, aKS, bKS,
aCVM, bCVM, aAD, bAD, nn, count, aZAD, bZAD
external kdis
integer NPAR, NSIG, MAXFN, IOPT, n, c, nn, count(4,5,14)
real kse(2), H(3), G(2), W(6), F, x(18), ablu, bblu, aKS, bKS,
B(18), D, Anc, Bnc, BB(18), aCVM, bCVM, aAD, bAD, aZAD, bZAD
double precision dseed

*** Enter the ZXMIN required constants
NPAR = Z
NSIG = 3
MAXFN = 500
IOPT = 0

*** Initialize the kse values to the blu estimates
kse(1) = ablu
kse(2) = bblu

*** Call ZXMIN to refine the kse values by minimizing
*** the Kolmogorov distance (KST) computed in the
*** subroutine KDIS
call ZXMIN(KDIS, NPAR, NSIG, MAXFN, IOPT, kse, H, G, F, W, IER)

*** Relabel the refined estimates of location and scale
aKS = kse(1)
bKS = kse(2)

*** Increment the KS counters
count(c, nn, 3) = count(c, nn, 3) + 1
count(c, nn, 4) = count(c, nn, 4) + 1
return
end
Subroutine KDIS(NPAR, kse, F)

---

c Purpose:  KDIS provides the function which is to be minimized
by ZXMIN for the Kolmogorov distance measure. The
location and scale parameters are altered to achieve
this minimization.

c Variables:  NPAR = number of parameters available to alter
c Inputs:  NPAR = number of parameters available to alter
          n = sample size
          kse = initial estimates (the blu estimates)
          x = array of ordered Pareto variates
          c = shape parameter

  c------------------------------------------------------------------
  c Outputs:  F = value of the function at the final estimates
             kse = revised estimates of location and scale;
                 these are the Kolmogorov minimum distance
                 estimates
  c------------------------------------------------------------------
  c Calculations:
  c          z(i) = 1 - (1 + (x(i)-a)/b)**(-c)
  c          DP(i) = ABS[ 1/n - z(i) ]
  c          DM(i) = ABS[ z(i) - (i-1)/n ]

  c *------------------------------------------------------------------
  c *** Variable Declarations:
  common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,
          aCVM,bCVM,aAD,bAD,nn,count,a2AD,b2AD
  integer NPAR,n,c,nn,count(4,5,14)
  real kse(NPAR),F,x(18),zi(18),DP(18),DM(18),DPLUS,
  1    DMINUS,KST,ablu,bblu,B(18),Anc,Bnc,BB(18),aKS,bKS,
  1    aCVM,bCVM,aAD,bAD,a2AD,b2AD
  double precision dseed
  c *** Calculate the Pareto cdf value [zi(j)] at each point
  c *** and the differences between the EDF step function
  c *** and the cdf points
  do 10 j=1,n
  10     z1(j) = 1-(1/(1+(x(j)-kse(1))/kse(2)))**(-c)
     DP(j) = ABS(j/real(n) - z1(j))
     DM(j) = ABS(z1(j) - (j-1)/real(n))
  continue
  c *** Select the maximum of the plus and minus differences
  DPLUS = MAX(DP(1),DP(2),DP(3),DP(4),DP(5),DP(6),DP(7)
          1 ,DP(8),DP(9),DP(10),DP(11),DP(12),DP(13),DP(14)
          1 ,DP(15),DP(16),DP(17),DP(18))
  DMINUS = MAX(DM(1),DM(2),DM(3),DM(4),DM(5),DM(6),DM(7)
          1 ,DM(8),DM(9),DM(10),DM(11),DM(12),DM(13),DM(14)

81
Select the maximum Kolmogorov distance measure and set $F$ equal to that distance. $F$ becomes the function which ZXMIN attempts to minimize by altering the values of the location and scale parameters.

$$KST = \max(DPLUS,DMINUS)$$

$F = KST$

return

Subroutine CVMMD

Purpose: CVMMD generates the minimum distance estimates of location and scale based upon minimizing the Cramer-von Mises distance measure defined in subroutine CVMDIS. This routine uses the blu estimates as the starting points for the estimate modifications.

Variables: $NPAR =$ number of parameters altered by minimizing the Cramer-von Mises (CVM) distance
$c$ $NSIG =$ number of significant digits for convergence
$c$ $MAXFN =$ maximum number of function evaluations
$c$ $IOPT =$ options selector (see IMSL manual on ZXMIN)
$c$ $H, G, W =$ vectors defined in IMSL manual on ZXMIN
$c$ $IER =$ error parameter (see IMSL manual on ZXMIN)
$c$ $F =$ value of CVM distance at the final parameter estimates
$c$ $cvme =$ CVM derived minimum distance estimates
$c$ $aCVM =$ CVM minimum distance location estimate
$c$ $bCVM =$ CVM minimum distance scale estimate
$c$ $ablu =$ blu estimate of location
$c$ $bblu =$ blu estimate of scale
$c$ $count =$ array of counters used to count the number of valid estimate values found
$c$

Inputs: $NPAR =$ number of parameters altered while minimizing
$c$ $NSIG =$ number of significant digits required
$c$ $MAXFN =$ maximum number of function evaluations
$c$ $IOPT =$ options selector (see IMSL manual on ZXMIN)
$c$ $cvme =$ initial estimates for the minimization process
$c$ $ablu =$ blu estimate of location
$c$ $bblu =$ blu estimate of scale
$c$

Outputs: $F =$ minimum value of the function being minimized
$c$ $cvme =$ revised estimate values
$c$ $aCVM =$ revised MD estimate of location $[aCVM=cvme(1)]$
$c$ $bCVM =$ revised MD estimate of scale $[bCVM=cvme(2)]$
$c$ $H, G, W =$ vectors defined in IMSL manual on ZXMIN
$c$ $IER =$ error parameter (see IMSL manual on ZXMIN)
$c$

Calculate: no calculations performed in this subroutine

82
### Variables Declaration:
- common n, x, c, ablu, bblu, dseed, B, D, Anc, Bnc, BB, aKS, bKS,
- $a_{CVM}, b_{CVM}, a_{AD}, b_{AD}, nn, count, a_{ZAD}, b_{ZAD}$
- external cvmdis
- integer NPAR, NSIG, MAXFN, IOPT, n, c, nn, count(4, 5, 14)
- real cvme(2), H(3), G(2), W(6), F, x(18), ablu, bblu, $a_{CVM}, b_{CVM}$
- B(18), D, Anc, Bnc, BB(18), aKS, bKS, $a_{AD}, b_{AD}, a_{ZAD}, b_{ZAD}$
- double precision dseed

### Enter the ZXMIN required constants
- $NPAR = 2$
- $NSIG = 3$
- $MAXFN = 500$
- $IOPT = 0$

### Initialize the cvme values to the blu estimates
- $cvme(1) = ablu$
- $cvme(2) = bblu$

### Call ZXMIN to refine the cvme values by minimizing
- the CVM distance ($W_Z$) computed in the
- subroutine CVMDIS
- call ZXMIN(CVMDIS, NPAR, NSIG, MAXFN, IOPT, cvme, H, G, F, W, IER)

### Relabel the refined estimates of location and scale
- $a_{CVM} = cvme(1)$
- $b_{CVM} = cvme(2)$

### Increment the CVM counters
- count($c, nn, 5) = count(c, nn, 5) + 1$
- count($c, nn, 6) = count(c, nn, 6) + 1$
- return

---

**Subroutine CVMDIS(NPAR, cvme, F)**

---

**Purpose:** CVMDIS provides the function which is to be minimized by ZXMIN for the Cramer-von Mises distance measure. The location and scale parameters are altered to achieve this minimization.

**Variables:**
- $NPAR =$ number of parameters available to alter
- $n =$ sample size
- $cvme =$ estimates of the parameters being altered
- $F =$ value of the function to be minimized
- $x =$ array of ordered Pareto variates
- $c =$ shape parameter
- $z1 =$ array of Pareto cdf points
- $ACV =$ the squared quantity in the $W_Z$ formula
- $SCVM =$ the sum of the $ACV$ quantities
- $W_Z =$ the CVM distance measure

**Inputs:**
- $NPAR =$ number of parameters available to alter
- $n =$ sample size
- $cvme =$ initial estimates (the blu estimates)
- $x =$ array of ordered Pareto variates
- $c =$ shape parameter
c Outputs:  
F = value of the function at the final estimates  
c cvme = revised estimates of location and scale;  
c these are the CVM minimum distance estimates  

c Calculations:  
\[ z(i) = 1 - (1 + [(x(i)-a)/b]^{(-c)}) \]  
\[ ACV(i) = (z(1) - (Z_1-1)/2n)^2 \]  
\[ SCVM = ACV(1) + ACV(2) + \ldots + ACV(n) \]  
\[ WZ = SCVM + 1/12n \]  

*** Variable Declarations:  
common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,  
aCVM,bCVM,aAD,bAD,nn,count,aAD,bAD  
integer NPAR,n,c,nn,count(4,5,14)  
real cvme(NPAR),F,x(18),zi(18),SCVM,ACV(18),WZ  
1 ablubblu,B(18),D,Anc,Bnc,BB(18),aKS,bKS,aCVM,bCVM,  
1 aAD,bAD,aAD,bAD  
double precision dseed  
SCVM = 0  
do 10 j=1,n  
zi(j) = 1 - (1/(1+(x(j)-cvme(1))/cvme(2)))^{(-c)}  
ACV(j) = (zi(j) - (Z_{j-1} - 1)/(2*real(n)))^{2}  
SCVM = SCVM + ACV(j)  
10 continue  
WZ = SCVM + (1/(12*real(n)))  
F = WZ  
return  
end  

Subroutine ADMO  

c Purpose: ADMO generates the minimum distance estimates of  
the location parameter based upon minimizing the  
Anderson-Darling distance measure defined in  
subroutine ADDIS. ADMO uses the blu estimates as the  
starting points for the estimate modifications.  

*** Variables: NPAR = number of parameters altered by minimizing  
the Anderson-Darling (A-D) distance  
NSIG = number of significant digits for convergence  
MAXFN = maximum number of function evaluations  
IOPT = options selector (see IMSL manual on ZMIN)  
H, G, W = vectors defined in IMSL manual on ZMIN  
IER = error parameter (see IMSL manual on ZMIN)  
F = value of A-D distance at the final  
parameter estimates  
ade = A-D derived minimum distance estimate
c aAD = A-D minimum distance location estimate
c ablu = blu estimate of location
c bblu = blu estimate of scale
c count = array of counters used to count the number of
valid estimate values found
c anda = array of calculated A-D distance measures
c esta = array of location estimates used to minimize
the A-D distance measure
icnt = counter for the number of estimates used

------------------------------------------------------------------

c Inputs: NPAR = number of parameters altered while minimizing
NSIG = number of significant digits required
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
ade = initial estimate for the minimization process
ablu = blu estimate of location
bblu = blu estimate of scale

c Outputs: F = minimum value of the function being minimized
ade = revised estimate values
aAD = revised MD estimate of location [aAD = ade(1)]
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)

------------------------------------------------------------------

c Calculate: no calculations performed in this subroutine

---

*** Variables Declaration:
common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,
aCVM,bCVM,aAD,bAD,nn,count,a2AD,b2AD,anda,esta,icnt
andb,estb,icntb,anda,esta,icnt,icntb,icntab
external addis
integer NPAR,NSIG,MAXFN,IOPT,n,c,nn,count(4,5,14),icnt,icntb
real ade(1),H(1),G(3),W(18),F,x(18),ablu,bblu,aAD,bAD
B(18),D,Anc,Bnc,BB(18),aKS,bKS,aCVM,bCVM,a2AD,b2AD
anda(500),esta(500),A01,anda(500),esta(500),ardab(500)
estaa(500),estbb(500)
double precision dseed

*** Enter the ZXMIN required constants
NPAR = 1
NSIG = 3
MAXFN = 500
IOPT = 0

*** Initialize the ade value to the blu estimate
ade(1) = ablu

*** Call ZXMIN to refine the ade values by minimizing
*** the Anderson-Darling distance (AD) computed in
*** the subroutine ADDIS
call ZXMIN(ADDIS,NPAR,NSIG,MAXFN,IOPT,ade,H,G,F,W,IER)
aAD = ade(1)

*** Reinitialize the icnt, anda, and esta arrays
icnt = 0
do 30 i = 1, n
   anda(i) = 0
   esta(i) = 0
30 continue
*** Increment AD counter for valid AD estimates
count(c,nn,7) = count(c,nn,7) + 1
return
end

Subroutine ADDIS(NPAR, ade, F)

Purpose: ADDIS provides the function which is to be minimized by ZXMIN for the Anderson-Darling distance measure.
The location parameter is altered to achieve the minimization.

Variables:  
- NPAR = number of parameters available to alter
- n = sample size
- ade = estimates of the parameter being altered
- F = value of the function to be minimized
- x = array of ordered Pareto variates
- c = shape parameter
- zi = array of Pareto cdf points
- AAA = array of terms to be summed in AD formula
- SAAD = sum of the AAA(i) terms
- AD = Anderson-Darling distance measure
- anda = array of calculated A-D distance measures
-esta = array of location estimates used to minimize the A-D distance measure
- icnt = counter for the number of estimates used

Inputs:  
- NPAR = number of parameters available to alter
- n = sample size
- ade = initial estimate (the blu estimate)
- x = array of ordered Pareto variates
- c = shape parameter

Outputs:  
- F = value of the function at the final estimate
- ade = revised estimate of location; this is the Anderson-Darling minimum distance estimate

Calculations:

$z(i) = 1 - (1 + [x(i)-a]/b)^{-c}$

$AAA(i) = (z_{i-1}) \left[ \ln z(1) + \ln (1-z(n+1-i)) \right]$  

$SAAD = AAA(1) + AAA(2) + \ldots + AAA(n)$

$AD = (-SAAD)/n - n$
*** Variable Declarations:
common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,
aCVM,bCVM,aAD,bAD,nn,count,a2AD,b2AD,anda,esta,icnt
,andb,estb,icntb,andab,estaa,estbb,icntab
integer NPAR,n,c,nn,count(4,5,14),icnt,icntb,icntab
real ade(NPAR),F,x(18),zi(18),AAA(18),SAAD,AD
1 ,ablu,bblu,B(18),D,Anc,Bnc,BB(18),aKS,bKS,aCVM,bCVM,
aAD,bAD,a2AD,b2AD,anda(500),esta(500)
1 ,andb(500),estb(500),ancab(500),estaa(500),estbb(500)
double precision dseed

do 10 j=1,n
  c *** Calculate the Pareto cdf point values
    zi(j) = 1-((1/(1+(x(j)-ade(1))/bblu))**c
  c *** Test zi(j) and [ 1 - zi(j) ] for negativity
    if (zi(j).le.0 .or. zi(j).ge.1) then
      go to 30
    end if
10 continue
  SAAD = 0
  c *** Calculate the Anderson-Darling distance
  do 20 m=1,n
    AAA(m) = (zm-1) * (log(zi(m)) + log(1-zi(n+1-m)))
    SAAD = SAAD + AAA(m)
20 continue
  AD = (-1) * (n + SAAD/n)
  c *** Save the AD and ade(1) values
    icnt = icnt + 1
    anda(icnt) = AD
    esta(icnt) = ade(1)
  c *** Relabel the A-D distance
    F = AD
    go to 40
30 ade(1) = esta(icnt-1)
40 return
end

Subroutine ADBMD

Purpose: ADBMD generates the minimum distance estimates of
the scale parameter based upon minimizing the
Anderson-Darling distance measure defined in
subroutine ADBDIS. ADBMD uses the blu estimates as the
starting points for the estimate modifications.

Variables: NPAR = number of parameters altered by minimizing
the Anderson-Darling (A-D) distance
NSIG = number of significant digits for convergence
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)
F = value of A-D distance at the final
parameter estimates

ade = A-D derived minimum distance estimate
bAD = A-D minimum distance scale estimate
ablu = blu estimate of location
bblu = blu estimate of scale
count = array of counters used to count the number of
valid estimate values found
andb = array of calculated A-D distance measures
estb = array of scale estimates used to minimize
the A-D distance measure
icntb = counter for the number of estimates used

Inputs:
NPAR = number of parameters altered while minimizing
NSIG = number of significant digits required
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
ade = initial estimate for the minimization process
ablu = blu estimate of location
bblu = blu estimate of scale

Outputs:
F = minimum value of the function being minimized
ade = revised estimate values
bAD = revised MD estimate of scale (bAD = ade(1))
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)

Calculate: no calculations performed in this subroutine

*** Variables Declaration:
common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,
1 aCVM,bCVM,aAD,bAD,nn,count,a2AD,b2AD anda,esta,icnt
1 ,andb,estb,icntb,andab,estaa,estbb,icntab
external adbdis
integer NPAR,NSIG,MAXFN,IOPT,n,c,nn,count(4,5,14),icnt
1 ,icntb,icntab,hhh
real ade(1),H(1),G(1),W(3),F,x(18),ablu,bblu,aAD,bAD
1 ,B(18),D,Anc,Bnc,BB(18),aKS,bKS,aCVM,bCVM,a2AD,b2AD anda
1 ,esta(500),esta(500),andb(500),estb(500),andb(500),
1 estaa(500),estbb(500),ADZ
double precision dseed

*** Enter the ZXMIN required constants
NPAR = 1
NSIG = 3
MAXFN = 500
IOPT = 0

*** Initialize the ade value to the blu estimate
ade(1) = bblu

*** Call ZXMIN to refine the ade values by minimizing
the Anderson-Darling distance (AD) computed in
the subroutine ABDIS

call ZXMIN(ABDIS,NPAR,NSIG,MAXFN,IOPT,ade,H,G,F,W,IER)
bPD = ade(1)

*** Reinitialize the icntb, andb, and estb arrays

25 icntb = 0

do 30 i = 1,w(z)
   andb(i) = 0
   estb(i) = 0

30 continue

*** Increment AD counter for valid AD estimates

count(c,nn,8) = count(c,nn,8) + 1

return

end

Subroutine ADBDIS(NPAR,ade,F)

C Purpose: ADBDIS provides the function which is to be minimized
by ZXMIN for the Anderson-Darling distance measure.
The scale parameter is altered to achieve the minimization.

C Variables: NPAR = number of parameters available to alter
n = sample size
ade = estimates of the parameter being altered
F = value of the function to be minimized
x = array of ordered Pareto variates
C = shape parameter
zi = array of Pareto 'f' points
AAA = array of terms to be summed in AD formula
SAAD = sum of the AAA(i) terms
AD = Anderson-Darling distance measure
andb = array of calculated A-D distance measures
estb = array of scale estimates used to minimize the A-D distance measure
icntb = counter for the number of estimates used

C Inputs: NPAR = number of parameters available to alter
n = sample size
ade = initial estimate (the blue estimate)
x = array of ordered Pareto variates
C = shape parameter

C Outputs: F = value of the function at the final estimate
ade = revised estimate of scale; this is the Anderson-Darling minimum distance estimate

C Calculations:

z(1) = 1 - (1 + [x(1)-a1/b]**(-c))

AAA(1) = (z(1)-1) * ln z(1) + ln (1-z(n+1-1))

SAAD = AAA(1) + AAA(2) + ... + AAA(n)

AD = (-SAAD)/n - n
*** Variable Declarations:

```c
common n,x,c,ablu,bblu,dseed,B,DO,Anc,Bnc,BB,aKS,bKS,
   aCVM,bCVM,aAD,bAD,nn,count,aZAD,bZAD,anda,esta,icnt
```

```c
integer NPAR,n,c,nn,count(4,5,14),icnt,icntb,icntab
real ade(NPAR),F,x(1B),zi(lB),AAA(I8),SAAD,AD
```

```c
double precision dseed
```

```c
do 10 j=l,n
   c * Calculate the Pareto cdf point values
   zi(j) = 1-(1/(1+(x(j)-ablu)/ade(j)))**c
```

```c
*** Test zi(j) and [ 1 - zi(j) ] for negativity
   if (zi(j).le.0 .or. zi(j).ge.l) then
      go to 30
   end if
```

```c
10 continue
```

```c
SAAD = 0
```

```c
*** Calculate the Anderson-Darling distance
```

```c
do ZO m=l,n
   AAA(m) = (Zm-1) * (log(zi(m)) + log(1-zi(n+1-m)))
   SAAD = SAAD + AAA(m)
```

```c
20 continue
```

```c
AD = (-1) * (n + SAAD/n)
```

```c
*** Save the AD and ade(i) values
```

```c
icntb - icntb + 1
```

```c
andb(icntb) - AD
```

```c
esta(icntb) - ade(i)
```

```c
*** Relabel the A-D distance
```

```c
F = AD
```

```c
go to 40
```

```c
30 ade(i) - esta(icntb) - j
```

```c
return
```

```c
end
```

Subroutine ADZMD

---

**Purpose:** ADZMD generates the minimum distance estimates of location and scale simultaneously, based on minimizing the Anderson-Darling distance measure defined in subroutine ADZDIS. ADZMD uses the blu estimates as the starting points for the estimate modifications.

**Variables:**
- NPAR - number of parameters altered by minimizing
- NSIG - number of significant digits for convergence
- IMAXFN - maximum number of function evaluations
- ILOPT - options selector (see IMSL manual on ZXMIN)

---

90
H, G, W - vectors defined in IMSL manual on ZXHIN
IER - error parameter (see IMSL manual on ZXHIN)
F - value of A-D distance at the final parameter estimates
ade - A-D derived minimum distance estimate
aZAD - A-D minimum distance location estimate
bZAD - A-D minimum distance scale estimate
ablu - blu estimate of location
bblu - blu estimate of scale
count - array of counters used to count the number of valid estimate values found
andab - array of calculated A-D distance measures
estaa - array of location estimates used to minimize the A-D distance measure
estbb - array of scale estimates used to minimize the A-D distance measure
icntab - counter for the number of estimates used

---
Inputs.
NPAR - number of parameters altered while minimizing
NSIG - number of significant digits required
MAXFN - maximum number of function evaluations
IOPT - options selector (see IMSL manual on ZXHIN)
ade - initial estimate for the minimization process
ablu - blu estimate of location
bblu - blu estimate of scale
---
Outputs.
F - minimum value of the function being minimized
ade - revised estimate values
aZAD - revised HD estimate of location (aZAD = ade(i));
bZAD - revised HD estimate of scale (bZAD = ade(i));
H, G, W - vectors defined in IMSL manual on ZXHIN
IER - error parameter (see IMSL manual on ZXHIN)
---
Calculate: no calculations performed in this subroutine
---

*** Variables Declaration.
common n,x,c,ablu,bblu,dseed,B,D,anc,anb,bb,aks,bks,
aCVH,bCVH,aAD,bAD,nn,count,aZAD,bZAD,and,esta,icnt,
anda,estb,icntb,andab,esta,estab,icntab
external adZdis
integer NPAR,NSIG,MAXFN,IOPT,n,c,nn,count(4,5,14)
,icnt,icntb,icntab,hhh
real ade(2),H(3),G(2),W(6),F,X(18),ablu,bblu,aAD,bAD
,bb(18),D,anc,anb,BB(18),aks,bks,aCVH,bCVH,aAD,bAD
,anda(500),esta(500),andb(500),estb(500),andab(500),
esta(500),estab(500),andb(500),nda3
double precision dseed

*** Enter the ZXHIN required constants
NPAR = 2
NSIG = 3
MAXFN = 500

91
** Initialize the ade value to the blu estimate

\[
\text{ade}(i) - \text{ablu} \\
\text{ade}(z) - \text{bblu}
\]

** Call ZXMIN to refine the ade values by minimizing

** the Anderson-Darling distance (AD) computed in

** the subroutine ADZDIS

\[
\text{call ZXMIN(ADZDIS,NPAR,NSIG,MAXFN,IOPT,ade,H,G,F,W,IER)}
\]

\[
aZAD = \text{ade}(1) \\
bZAD = \text{ade}(z)
\]

** Reinitialize the icntab, andab, estbb, and estaa arrays

\[
do \ 30 \ i = 1,w(z) \\
\quad \text{andab}(i) = 0 \\
\quad \text{estaa}(i) - 0 \\
\quad \text{estbb}(i) - 0
\]

30 \ continue

** Increment AD counter for valid AD estimates

\[
\text{count(c,nn,9)} = \text{count(c,nn,9)} + 1 \\
\text{count(c,nn,10)} = \text{count(c,nn,10)} + i
\]

** Return

** Subroutine ADZDIS(NPAR,ade,F)

---

** Purpose: ** ADZDIS provides the function which is to be minimized

** by ZXMIN for the Anderson-Darling distance measure.

** The location and scale parameters are both altered to

** achieve the minimization.

** Variables: **

NPAR = number of parameters available to alter

n = sample size

ade = estimates of the parameter being altered

F = value of the function to be minimized

x = array of ordered Pareto variates

c = shape parameter

zi = array of Pareto cdf points

AAA = array of terms to be summed in AD formula

SAAO = sum of the AAA(1) terms

AD = Anderson-Darling distance measure

andab = array of calculated A-D distance measures

estaa = array of location estimates used to minimize

the A-D distance measure

estbb = array of scale estimates used to minimize

the A-D distance measure

icntab = counter for the number of estimates used

** Inputs: **

NPAR = number of parameters available to alter

n = sample size

ade = initial estimate (the blu estimate)

x = array of ordered Pareto variates

c = shape parameter
c Outputs: F = value of the function at the final estimate
ade = revised estimates of location and scale;
these are the Anderson-Darling minimum
distance estimates

c Calculations:
  z(i) = 1 - (1 + [x(i)-a]/b)**(-c)
  AAA(i) = (Zi-1) [ ln z(i) + ln ( 1-z(n+1-i) ) ]
  SAAD = AAA(1) + AAA(2) + ... + AAA(n)
  AD = (-SAAD)/n - n

c----------------------------------------------------------------------
c*** Variable Declarations:
common n,x,c,ablu,bblu,dseed,B,D,anc,bn,bb,aKS,bKS,
   aCVM,bCVM,aAD,bAD,nn,count,a2AD,b2AD,anda,esta,icnt
   andb,estb,icntb,estab,icntab
integer NPAR,n,c,nn,count(4,5,14),icnt,icntb,icntab
real ade(NPAR),F,x(18),zi(18),AAA(18),SAAD,AD
   ,ablu,bblu,B(18),D,anc,bn,bb(18),aKS,bKS,aCVM,bCVM,
   aAD,bAD,a2AD,b2AD,anda(500),esta(500),andb(500),
   estb(500),andab(500),estaa(500),estbb(500)
double precision dseed
do 10 j=1,n
  *** Calculate the Pareto cdf point values
  zi(j) = 1-1/(1/(1+(x(j)-ade(1))/ade(2)))**c
  *** Test zi(j) and [ 1 - zi(j) ] for negativity
  if (zi(j).le.0 .or. zi(j).ge.1) then
    go to 30
  end if
 10 continue
SAAD = 0
  *** Calculate the Anderson-Darling distance
do 20 m=1,n
  AAR(m) = (2*m-1) * (log(zi(m)) + log(1-z1(n+1-m)))
  SAAD = SAAD + AAR(m)
20 continue
AD = (-1) * (n + SAAD/n)
  *** Save the AD and ade(1) values
  icntab = icntab + 1
  andab(icntab) = AD
  esta(icntab) = ade(1)
  estab(icntab) = ade(2)
  *** Relabel the A-D distance
  F = AD
  go to 40
30 ade(1) = estaa(icntab-1)
ade(2) = estbb(icntab-1)
40 return
end
Subroutine CVAMD

Purpose: CVAMD generates the minimum distance estimates of the location parameter based upon minimizing the Cramér-von Mises distance measure defined in subroutine CVADIS. CVAMD uses the Blu estimates as the starting points for the estimate modifications.

Variables: NPAR = number of parameters altered by minimizing the Cramér-von Mises (CVM) distance
NSIG = number of significant digits required
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)
F = value of CVM distance at the final parameter estimates
cvme = CVM derived minimum distance estimate
alCV = CVM minimum distance location estimate
ablu = Blu estimate of location
bblu = Blu estimate of scale
count = array of counters used to count the number of valid estimate values found

Inputs: NPAR = number of parameters altered while minimizing
NSIG = number of significant digits required
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
cvme = initial estimate for the minimization process
ablu = Blu estimate of location
bblu = Blu estimate of scale

Outputs: F = minimum value of the function being minimized
cvme = revised estimate values
alCV = revised MD estimate of location [alCV=cvme(1)]
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)

Calculate: no calculations performed in this subroutine

*** Variables Declaration:
common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,
aCVM,bCVM,aAD,bAD,nn,count,aZAD,bZAD,anda,esta,icnt
1 ,andb,estb,icntb,andab,esta,estab,icntab
1 ,alCV,bICVM,alKS,bIKS
external CVADIS
integer NPAR,NSIG,MAXFN,IOPT,n,c,nn,count(4,5,14),icnt,hh
1 ,icntb,icntab
real cvme(1),H(1),G(1),W(3),F,x(18),ablu,bblu,aAD,bAD
1 ,B(18),D,Anc,Bnc,BB(18),aKS,bKS,aCVM,bCVM,aZAD,bZAD
1 ,anda(500),esta(500),A01,andb(500),estb(500),andab(500)
double precision dseed

*** Enter the ZXMIN required constants
NPAR = 1
NSIG = 3
MAXFN = 500
IOPT = 0

*** Initialize the cvme value to the blu estimate
cvme(1) = ablu

*** Call ZXMIN to refine the cvme values by minimizing
*** the Cramer-von Mises distance (W2) computed in
*** the subroutine CVADIS
call ZXMIN(CVADIS,NPAR,NSIG,MAXFN,IOPT,cvme,H,G,F,U,IER)

*** Relabel the refined estimates of location
alCV = cvme(1)

*** Increment the alCV counter
count(c,nn,11) = count(c,nn,11) + 1
return
end

Subroutine CVADIS(NPAR,cvme,F)

---

Subroutine CVADIS provides the function which is to be minimized
by ZXMIN for the Cramer-von Mises distance measure.
The location parameter is altered to
achieve this minimization.

---

Variables:

NPAR = number of parameters available to alter
n = sample size
cvme = estimates of the parameters being altered
F = value of the function to be minimized
x = array of ordered Pareto variates
c = shape parameter
zi = array of Pareto cdf points
ACV = the squared quantity in the W2 formula
SCVM = the sum of the ACV quantities
W2 = the CVM distance measure

---

Inputs:
NPAR = number of parameters available to alter
n = sample size
cvme = initial estimates (the blu estimates)
x = array of ordered Pareto variates
c = shape parameter

---

Outputs:
F = value of the function at the final estimates
cvme = revised estimates of location and scale;
these are the CVM minimum distance estimates

---

Calculations:

z(1) = 1 - (1 + (x(1)-al/b)**(-c))

ACV(1) = [ z(1) - (zi-1)/Zn ]**2
SCVM = ACV(1) + ACV(2) + ... + ACV(n)

WZ = SCVM + 1/IZn

*** Variable Declarations:

**common nx,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,**
  1       acVM,bcVM,aAD,bAD,nn,count,aZAD,bZAD,anda,esta,icnt
  1       ,andb,estb,icntb,andab,estaa,estbb,icntab
1       ,aICV,bICV,aIKS,bIKS
1       integer NPAR,NSIG,MAXFN,IOPT,n,c,nn,count(4,5,14),icnt,ih
1     ,icntb,icntab
1       real cvme(1),H(1),G(1),W(1),F,x(18),ablu,bblu,aAD,bAD
1     ,B(18),D,Anc,Bnc,BB(18),aKS,bKS,acVM,bcVM,aZAD,bZAD
1     ,anda(500),esta(500),estb(500),aICV,bICV,aIKS,bIKS
1     ,zi(18),SCVM,AC(V(18),WZ)

double precision dseed

SCVM = 0

do 10 j=1,n
  zi(j) = 1-(1/(1+(x(j)-cvme(1))/bblu))**c
  ACV(j) = (zi(j) - (Z*(j-1)/(Z*real(n))))**2
  SCVM = SCVM + ACV(j)
10 continue

WZ = SCVM + (1/(IZ*real(n)))

F = WZ

return

end

Subroutine CVBMD

Purpose: CVBMD generates the minimum distance estimates of
the scale parameter based upon minimizing the
Cramer-von Mises distance measure defined in
subroutine CVBDIS. CVBMD uses the blu estimates as the
starting points for the estimate modifications.

Variables: NPAR = number of parameters altered by minimizing
the Cramer-von Mises (CVM) distance
NSIG = number of significant digits for convergence
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)
F = value of CVM distance at the final
parameter estimates
cvme = CVM derived minimum distance estimate
bICV = CVM minimum distance scale estimate
ablu = blu estimate of location
bblu = blu estimate of scale
count = array of counters used to count the number of
valid estimate values found

Inputs: NPAR = number of parameters altered while minimizing
NSIG = number of significant digits required
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
cvme = initial estimate for the minimization process
abl = blu estimate of location
bbl = blu estimate of scale

Outputs: F = minimum value of the function being minimized
cvme = revised estimate values
b1CV = revised MD estimate of scale [b1CV=cvme(1)]
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)

Calculate: no calculations performed in this subroutine

### Variables Declaration:

```c
common n,x,c,abl,bblu,dseed,B,0, Anc, Bnc, BB, aKS, bKS,
    aCVM,bCVM,aAD,bAD,nn, count, aZAD,bZAD, anda, esta, icnt
    , andb, estb, icntb, andab, estaab, estbb, icntab
    aCVM,bCVM,aZAD,bZAD

external CVBDIS
integer NPAR,NSIG,MAXFN,IOPT,n,c,nn,count(4,5,14),icnt,hh
    icntb,icntab
real cvme(l),H(1),G(1),W(3),F,x(18),abl,bblu,aAD,bAD
    ,B(18),D, Anc, Bnc, BB(l8), aKS, bKS, aCVM,bCVM,aZAD,bZAD
    andb(500),esta(500),AD1, andab(500),estb(500), andab(500)
    estaa(500),estbb(500),aICV,bICV,aKS,bKS
double precision dseed
```

### Enter the ZXMIN required constants
NPAR = 1
NSIG = 3
MAXFN = 500
IOPT = 0

### Initialize the cvme value to the blu estimate

cvme(1) = bblu

### Call ZXMIN to refine the cvme values by minimizing

```c
call ZXMIN(CVBDIS,NPAR,NSIG,MAXFN,IOPT,cvme,H,G,F,W,IER)
```

### Relabel the refined estimates of scale
b1CV = cvme(1)

### Increment the b1CV counter

count(c,nn,12) = count(c,nn,12) + 1

```c
return
end
```

Subroutine CVBDIS(NPAR,cvme,F)

Purpose: CVBDIS provides the function which is to be minimized

97
by ZXMIN for the Cramer-von Mises distance measure.
The scale parameter is altered to achieve this minimization.

Variables: 
- \( NPAR \) = number of parameters available to alter
- \( n \) = sample size
- \( \text{cvme} \) = estimates of the parameters being altered
- \( F \) = value of the function to be minimized
- \( x \) = array of ordered Pareto variates
- \( c \) = shape parameter
- \( z_i \) = array of Pareto cdf points
- \( \text{ACV} \) = the squared quantity in the WZ formula
- \( \text{SCVM} \) = the sum of the ACV quantities
- \( \text{UZ} \) = the CVM distance measure

Inputs: 
- \( NPAR \) = number of parameters available to alter
- \( n \) = sample size
- \( \text{cvme} \) = initial estimates (the blu estimates)
- \( x \) = array of ordered Pareto variates
- \( c \) = shape parameter

Outputs: 
- \( F \) = value of the function at the final estimates
- \( \text{cvme} \) = revised estimates of scale; these are the CVM minimum distance estimates

Calculations:
\[
z(i) = 1 - (1 + (x(i)-\text{ablu})/\text{cvme(1)})^{*(-c)}
\]
\[
\text{ACV}(i) = [ z(i) - (Zi-1)/Zn ]**2
\]
\[
\text{SCVM} = \text{ACV}(1) + \text{ACV}(2) + \ldots + \text{ACV}(n)
\]
\[
\text{WZ} = \text{SCVM} + 1/12n
\]

Variable Declarations:
- \( \text{cvme(1)}, H(1), G(1), W(3), F, x(18), \text{ablu, bblu, aAD, bAD} \)
- Integer \( \text{NN, c, count(4, 5, 14, i, c, cnt, hh)} \)
- Real \( \text{cvme(1), H(1), G(1), W(3), F, x(18), ablu, bblu, aAD, bAD} \)
- Double precision \( \text{dseed} \)

\[
\text{SCVM} = 0
\]
do 10 \( j = 1, n \)
\[
zj(j) = 1 - (1/(1+(x(j)-\text{ablu})/\text{cvme(1)}))^{*(-c)}
\]
ACV(j) = (zi(j) - (Z*j-1)/(Z*real(n)))**Z
SCVM = SCVM + ACV(j)
10 continue
W2 = SCVM + (1/(1Z*real(n)))
F = W2
return
end
Subroutine KSAMD

Purpose: KSAMD generates the minimum distance estimates of
location based upon minimizing the
Kolmogorov distance measure defined in subroutine
KADIS. This routine uses the blu estimates as the
starting points for the estimate modifications.

Variables: NPAR = number of parameters altered by minimizing
the Kolmogorov distance
NSIG = number of significant digits for convergence
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)
F = value of Kolmogorov distance at the final
parameter estimates
kse = Kolmogorov derived minimum distance estimates
alks = Kolmogorov minimum distance location estimate
ablu = blu estimate of location
bblu = blu estimate of scale
count = array of counters used to count the number of
valid estimate values found

Inputs: NPAR = number of parameters altered while minimizing
NSIG = number of significant digits required
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
ks = initial estimates for the minimization process
ablu = blu estimate of location
bblu = blu estimate of scale

Outputs: F = minimum value of the function being minimized
kse = revised estimate values
alks = revised MD estimate of location [alks = kse(1)]
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)

Calculate: no calculations performed in this subroutine

Variables Declaration:

common n, x, c, ablu, bblu, dseed, B, D, Anc, Bnc, BB, aks, bks,
aCVM, bCVM, aAD, bAD, nn, count, aZAD, bZAD, anda, esta, icnt
andb, estb, icntb, andab, estaa, estbb, icntab
**Subroutine KADIS**

**Purpose:** KADIS provides the function which is to be minimized by ZXMIN for the Kolmogorov distance measure. The location parameter is altered to achieve this minimization.

**Variables:**
- **NPAR** = number of parameters available to alter
- **n** = sample size
- **kse** = estimates of the parameters being altered
- **F** = value of the function to be minimized
- **x** = array of ordered Pareto variates
- **c** = shape parameter
- **zi** = array of Pareto cdf points
- **DP** = positive differences between the EDF and cdf
- **DM** = negative differences between the EDF and cdf
- **DPLUS** = maximum positive difference
- **DMINUS** = maximum negative difference
- **KST** = Maximum of DPLUS and DMINUS

**Inputs:**
- **NPAR** = number of parameters available to alter
- **n** = sample size
- **kse** = initial estimates (the blu estimates)
- **x** = array of ordered Pareto variates
c shape parameter

Outputs:  
  F = value of the function at the final estimates
  kse = revised estimates of location; these are the Kolmogorov minimum distance estimates

Calculations:
  \[ z(i) = 1 - \left( 1 + \frac{x(i) - a}{b} \right)^{-c} \]

\[ DP(i) = \text{ABS} \left( \frac{i}{n} - z(i) \right) \]

\[ DM(i) = \text{ABS} \left( z(i) - \frac{(i-1)}{n} \right) \]

** Variable Declarations:

```plaintext
common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,
  aCVM,bCVM,aAD,bAD,nn,count,a2AD,b2AD,anda,esta,icnt
  ,aICV,bICV,aKS,bKS

integer NPAR,NSIG,MAXFN,IOPT,n,c,nn,count(4,5,14),icnt,hh
  ,icntb,icntab
real kse(1),K(1),G(1),W(3),F,x(18),ablu,bblu,aAD,bAD
  ,B(18),D,Anc,Bnc,BB(18),aKS,bKS,aCVM,bCVM,a2AD,b2AD
  ,anda(500),esta(500),AD1,andb(500),estb(500),anda(500)
  ,estaa(500),estb(500),aICV,bICV,aKS,bKS
  ,zi(18),DP(18),DM(18),DPLUS,DMINUS,KST

double precision dseed
```

** Calculate the Pareto cdf value \( \text{zi}(j) \) at each point
** and the differences between the EDF step function
** and the cdf points

\[ \text{zi}(j) = \frac{1-(1/(1+(x(j)-kse(1))/bblu))^c}{\text{real}(n) - \text{zi}(j)} \]
\[ DP(j) = \text{ABS}(\text{zi}(j) - \text{real}(n) / \text{real}(n)) \]
\[ DM(j) = \text{ABS}(\text{zi}(j) - (j-1)/\text{real}(n)) \]

** Select the maximum of the plus and minus differences

\[ \text{DPLUS} = \text{MAX}(DP(1),DP(2),DP(3),DP(4),DP(5),DP(6),DP(7)) \]
\[ \text{DPLUS} = \text{MAX}(DP(8),DP(9),DP(10),DP(11),DP(12),DP(13),DP(14)) \]
\[ \text{DPLUS} = \text{MAX}(DP(15),DP(16),DP(17),DP(18)) \]
\[ \text{DMINUS} = \text{MAX}(DM(1),DM(2),DM(3),DM(4),DM(5),DM(6),DM(7)) \]
\[ \text{DMINUS} = \text{MAX}(DM(8),DM(9),DM(10),DM(11),DM(12),DM(13),DM(14)) \]
\[ \text{DMINUS} = \text{MAX}(DM(15),DM(16),DM(17),DM(18)) \]

** Select the maximum Kolmogorov distance measure and
** set F equal to that distance. F becomes the
** function which ZKMIN attempts to minimize by
** altering the values of the location parameter

\[ KST = \text{MAX}(\text{DPLUS},\text{DMINUS}) \]

F = KST

return

end

101
Subroutine KSBMD

Purpose: KSBMD generates the minimum distance estimates of scale based upon minimizing the Kolmogorov distance measure defined in subroutine KBDIS. This routine uses the blu estimates as the starting points for the estimate modifications.

Variables: NPAR = number of parameters altered by minimizing the Kolmogorov distance
NSIG = number of significant digits for convergence
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)
F = value of Kolmogorov distance at the final parameter estimates
kse = Kolmogorov derived minimum distance estimates
bIKS = Kolmogorov minimum distance scale estimate
ablu = blu estimate of location
bblu = blu estimate of scale
count = array of counters used to count the number of valid estimate values found

Inputs: NPAR = number of parameters altered while minimizing
NSIG = number of significant digits required
MAXFN = maximum number of function evaluations
IOPT = options selector (see IMSL manual on ZXMIN)

kse = initial estimates for the minimization process
ablu = blu estimate of location
bblu = blu estimate of scale

Outputs: F = minimum value of the function being minimized
kse = revised estimate values
bIKS = revised MD estimate of scale (\textit{bIKS} = \textit{kse}(l))
H, G, W = vectors defined in IMSL manual on ZXMIN
IER = error parameter (see IMSL manual on ZXMIN)

Calculate: no calculations performed in this subroutine

Variables Declaration:

common n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,
1 aCVM,bCVM,aAD,bAD,nn,count,a2AD,b2AD,anda,esta,icnt
1 ,icntb,icntab
1 ,aCVM,bCVM,aZAD,bZAD,anda,esta,estab,icntab
1 ,aCVM,bCVM,aIKS,bIKS

external KBDIS

integer NPAR,NSIG,MAXFN,IOPT,n,c,nn,count(4,5,14),icnt,nn
1 ,icntb,icntab
real kse(1),H(1),G(1),W(3),F,x(18),ablu,bblu,aAD,bAD
1 ,b(18),D,Anc,Bnc,BB(18),aKS,bKS,aCVM,bCVM,a2AD,b2AD
1 ,anda(500),esta(500),AD1,andb(500),estb(500),andab(500)
Subroutine KBDIS(NPAR,kse,F)

Purpose: KBDIS provides the function which is to be minimized by ZXMIN for the Kolmogorov distance measure. The scale parameter is altered to achieve this minimization.

Variables:
- NPAR = number of parameters available to alter
- n = sample size
- kse = estimates of the parameters being altered
- F = value of the function to be minimized
- x = array of ordered Pareto variates
- c = shape parameter
- zi = array of Pareto cdf points
- DP = positive differences between the EDF and cdf
- DM = negative differences between the EDF and cdf
- DPLUS = maximum positive difference
- DMINUS = maximum negative difference
- KST = Maximum of DPLUS and DMINUS

Inputs:
- NPAR = number of parameters available to alter
- n = sample size
- kse = initial estimates (the blu estimates)
- x = array of ordered Pareto variates
- c = shape parameter

Outputs:
- F = value of the function at the final estimates
- kse = revised estimates of scale; these are the Kolmogorov minimum distance estimates
Calculations:
\[ z(i) = 1 - \left( 1 + \frac{x(i) - a}{b} \right)^{-c} \]

\[ DP(i) = \text{ABS} \left( \frac{i}{n} - z(i) \right) \]

\[ DM(i) = \text{ABS} \left( z(i) - \frac{(i-1)}{n} \right) \]

**Variable Declarations:**
```
commom n,x,c,ablu,bblu,dseed,B,D,Anc,Bnc,BB,aKS,bKS,
aCVM,bCVM,aAD,bAD,nn,count,a2AD,b2AD,anda,esta,icnt
1   ,andb,estb,icntb,andab,estaa,estbb,icntab
1   ,aICV,bICV,aIKS,bIKS
integer NPAR,NSIG,MAXFN,IOPT,n,c,nn,count(4,5,14),icnt,hh
1   ,icntb,icntab
real kse(i),H(1),G(1),W(3),Fx(IB),ablu,bblu,aAD,bAD
1   ,B(18),D,Anc,Bnc,BB(18),aKS,bKS,aCVM,bCVM,a2AD,b2AD
1   ,anda(500),esta(500),AD1,anda(500),estb(500),andab(500)
1   ,estaa(500),estbb(500),aICV,bICV,aIKS,bIKS
1   ,zi(18),DP(18),DM(18),OPLUS,DMINUS,KST
double precision dseed
```

*** Calculate the Pareto cdf value \([z_i(j)]\) at each point
*** and the differences between the EDF step function
*** and the cdf points
```
do 10 j=1,n
   zi(j) = 1-(1/(1+(x(j)-ablu)/'kse(1))))**c
   DP(j) = \text{ABS}(j/\text{real}(n) - zi(j)),
   DM(j) = \text{ABS}(zi(j) - (j-1)/\text{real}(n))
10 continue
```

*** Select the maximum of the plus and minus differences
```
OPLUS = \text{MAX}(DP(1),DP(2),DP(3),DP(4),DP(5),DP(6),DP(7))
1   ,DP(8),DP(9),DP(10),DP(11),DP(12),DP(13),DP(14)
1   ,DP(15),DP(16),DP(17),DP(18))
DMINUS = \text{MAX}(DM(1),DM(2),DM(3),DM(4),DM(5),DM(6),DM(7))
1   ,DM(8),DM(9),DM(10),DM(11),DM(12),DM(13),DM(14)
1   ,DM(15),DM(16),DM(17),DM(18))
```

*** Select the maximum Kolmogorov distance measure and
*** set F equal to that distance. F becomes the
*** function which ZXMIN attempts to minimize by
*** altering the values of the location parameter
KST = \text{MAX}(OPLUS,DMINUS)
F = KST
return
end
BIBLIOGRAPHY


Vita

Major Dennis J. Charek was born on 21 December 1949 in Cleveland, Ohio. He graduated from Revere High School in Richfield, Ohio in 1967 and attended Kent State University from which he received the degree of Bachelor of Science in Physics in 1971. Upon graduation, he enlisted in the USAF in the accounting and finance specialty area. He received his commission in 1973 through the Officer Training School program. He was first employed as a Space Surveillance Officer at Detachment 7, 14th Missile Warning Squadron, MacDill AFB, FL in 1973. He then went overseas in 1977 to the 13th Missile Warning Squadron, Clear AFS, AK to serve as a Space Surveillance Officer. In 1978, he was called to serve a tour at the Aerospace Defense Command Headquarters at Peterson AFB, CO as the SLBM Radar Space Operations Officer. In 1980, he moved to Strategic Air Command Headquarters, Offutt AFB, NE where he was the Ballistic Missile Early Warning System Radar Manager. During this tour, he earned a Master of Science in Systems Management degree from the University of Southern California. He then moved to the Foreign Technology Division, Wright-Patterson AFB, OH in 1983 to serve as a Radar Development Analyst. He then entered the School of Engineering, Air Force Institute of Technology in 1984.

Permanent Address: 4691 Sunset Drive
Richfield, Ohio 44286
Title: A COMPARISON OF ESTIMATION TECHNIQUES FOR THE THREE PARAMETER PARETO DISTRIBUTION

Thesis Advisor: Dr. Albert H. Moore, Professor
This thesis compared minimum distance estimation with best linear unbiased estimation to determine which technique provided the most accurate estimates for location and scale parameters as applied to the three parameter Pareto distribution. Two minimum distance estimators were developed for each of the three distance measures used (Kolmogorov, Cramer-von Mises, and Anderson-Darling) resulting in six new estimators. For a given sample size and shape parameter, the location and scale parameters were estimated. Varying combinations of sample sizes (6, 9, 12, 15, 18) and shape parameters (1, 2, 3, 4) were tested.

A Monte Carlo methodology was used to generate the 1000 sample sets of Pareto random variates for each sample size-shape parameter combination with location and scale parameters both set equal to 1. The best linear unbiased estimator and the six minimum distance estimators provided parameter estimates based on each sample set. Finally, the parameter estimates were compared using the mean square error as the evaluation tool. The results of the research indicate that the best linear unbiased estimation technique provided more accurate estimates of location and scale for the three parameter Pareto distribution than did the minimum distance estimation techniques.