BOX-COX BOOTSTRAP AND CLASSICAL INFERENCE METHODS FOR THE SHAPE PARAMETER OF THE BURR TYPE X DISTRIBUTION

by

TRACEY K. HARRIS, B.S.

A THESIS

IN

STATISTICS

Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
the Requirements for
the Degree of

MASTER OF SCIENCE

Approved

____
Chairperson of the Committee

____

Accepted

____
Dean of the Graduate School

May, 2003

ACKNOWLEGEMENTS

I would like to thank all the people that helped me put this together, my family, friends, and my advisor Dr. James Surles. Without everyone's help and guidance, I would not have been able to have the strength to finish what I started.

TABLE OF CONTENTS

ABSTRACT

In this thesis, three methods for inference for the Burr Type X distribution shape parameter are presented. Some background on the Burr Type X distribution is given and the methods used for inference are also described within this thesis. A simulation study is conducted to assess the performance of the three methods, and their performances are compared.

LIST OF TABLES

CHAPTER I

INTRODUCTION

The purpose of this thesis is to examine inference procedures for the shape parameter from a Burr type X distribution (referred to here as the Burr X). The general procedure typically followed in this situation is to first identify an estimate of the parameter for which inference is desired. The sampling distribution of this estimate is identified or approximated, and inference procedures are developed using this sampling distribution. Unfortunately, in this case, the sampling distribution has not been identified exactly or approximately, which so often happens. For the Burr type X, the maximum likelihood estimate (m.l.e.) of the shape parameter has a normal distribution as the sample size tends to infinity $(n \to \infty)$. It is sometimes the case, however, that using the limiting distribution as an approximation provides poor inference, even for large sample sizes. This thesis will examine the effectiveness of an inference procedure that assumes the distribution to be exactly normal, regardless of sample size, with a method of combining bootstrap techniques and the Box-Cox class of transformations to hopefully improve inference for these situations. This latter technique was described by Surles (1999) and a related procedure was described by Chen and Loh (1991), and is described briefly next.

Suppose that θ is an unknown parameter of interest. In this thesis the estimate for θ is considered to be the m.l.e. $\hat{\theta}$. It will also be assumed that the distribution of $\frac{(\hat{\theta} - \theta)}{s}$ converges to standard normal as the sample size tends to infinity $(n \to \infty)$, where

s is some estimate of the standard deviation of the sampling distribution of $\hat{\theta}$. The basic idea given by Surles (1999) is that there should be a transformation h for which $h(\hat{\theta})$ is approximately normal. The task, then, is to identify h. In this thesis, h will be a Box-Cox power transformation, and will be estimated through the use of bootstrap methods.

Bootstrapping was introduced by Efron (1979) as a method for estimation of standard errors. The bootstrap method is primarily computer based, and the idea is that the bootstrap sample is a randomly selected re-sampled data set from the original data set $X_1, \ldots, X_n$ with replacement. With the bootstrap sample we can estimate the sampling distribution of the estimator. In this thesis, the sampling distribution will be estimated using standard parametric and non-parametric bootstrap methods, and a transformation will be found for which the sampling distribution of the transformed estimator is as "close" to normal as possible. The transformation of interest is the power transformation introduced by Box and Cox (1964), referred to here as the Box-Cox transformation.

The Box-Cox transformation is used to provide homogeneity and normality to a model. The original objective of this transformation was to simultaneously achieve three things:

1. To achieve normality of distributions,
2. To achieve linearity within normal models,
3. To achieve constancy of error variance.

The Box-Cox transformation will be modified here so that the transformed data follows a more normal distribution than the original data. This transformation was applied by

Surles (1999) in order to obtain an improved lower bound for R which is defined as $P(Y<X)$, where X and Y are independent Burr type X random variables.

Within this thesis, confidence intervals for the parameter will be used to assess the effectiveness of the inference procedures. Simulation studies will be conducted in which coverage probabilities will be estimated and compared to the ideal coverage probability and to coverage probabilities for other existing inference procedures. Several authors have studied inference for various quantities from the Burr type X, but to date no one has investigated inference for the parameters that define the Burr type X.

In Chapter II, the method of combining the bootstrap with Box-Cox transformations is discussed in detail. In Chapter III, the Burr type X distribution is introduced, and exact inference for the shape parameter is described for a special case. The results of the simulation study are given in Chapter IV.

CHAPTER II

BOOTSTRAPPING AND THE BOX-COX TRANSFORMATION

Bootstrapping is carried out by having an original data set $X_1,\ldots,X_n$, and re-sampling from an estimate of the cumulative distribution function (c.d.f.) of $X_1,\ldots,X_n$ such that there are B re-sampled data sets. The re-sampled data set will be denoted as $X_i^* = \left(X_{i1}^*, X_{i2}^*,\ldots,X_{in}^*\right)$, $i = 1\ldots B$. Inferences for the quantity $g=g(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is the vector of parameters, generally employ a test statistic, denoted as $\hat{g} = T\left(X_1, X_2,\ldots,X_n\right)$. In order to estimate the sampling distribution of $\hat{g}$, two methods are employed, the nonparametric and parametric bootstrap methods (Davison and Hinkley, 1997). An algorithm to describe the bootstrap method is as follows:

1. Draw B independent bootstrap samples $X_1^*, X_2^*,\ldots,X_B^*$ with replacement, each of size n from some estimate of the c.d.f. of $X_1,\ldots,X_n$.
2. Calculate the bootstrap replication of $\hat{g}_i^* = T\left(X_{i1}^*, X_{i2}^*,\ldots,X_{in}^*\right)$, $i = 1,\ldots B$, for each bootstrap sample.
3. Estimate the standard error of $\hat{g}$ by $se_B = \left\{\dfrac{\sum_{i=1}^{B}[\hat{g}_i^* - \hat{g}^*]^2}{B-1}\right\}^{\frac{1}{2}}$ where $\hat{g}^* = \dfrac{\sum_{i=1}^{B}\hat{g}_i^*}{B}$.

The parametric bootstrap method involves having a mathematical model whose parameters that completely determine the probability density function (p.d.f.)

of $X_1,\ldots,X_n$. When this method is used, a distribution is assumed for the sample and then the algorithm described above is used, where the estimate of the c.d.f. is found parametrically. For example, in the Burr X case $F(x \mid \hat{\theta},\hat{\sigma})$ could be used as the estimate of the c.d.f., where $\hat{\sigma}$ and $\hat{\theta}$ are the m.l.e.'s for σ and θ, and $F(x \mid \theta,\sigma) = [1 - e^{-(x/\sigma)^2}]^{\theta}$ is the Burr X c.d.f. The nonparametric method is used when there is not an explicitly given mathematical model to use, but it is assumed that the re-sampled data sets are independently and identically distributed (i.i.d.). When the nonparametric method is used the analyst is alleviated from making any parametric assumptions about the population. With the parametric method, its advantage is that of providing more accurate answers if the parametric model is appropriate.

After the original data is re-sampled and the bootstrap replications of $\hat{g}$ are obtained, $(\hat{g}_1^*, \hat{g}_2^*, \ldots, \hat{g}_B^*)$, they are transformed so that the estimated sampling distribution is as close to normal as possible. The transformation used here is the Box-Cox transformation. Box and Cox (1964) proposed a method of transformations in order to attain a close-to-normal distribution, with constant variance and a simple model structure in regression models. Box and Cox (1964) considered the parametric family of transformations for the one-parameter case given by

$$h(X,\lambda) = \begin{cases} \dfrac{X^{\lambda} - 1}{\lambda}, \lambda \neq 0 \\ \log(X), \lambda = 0 \end{cases}, \quad \text{for } X > 0, \tag{2.1}$$

for improving the validity of the normal theory linear model (Hernandez and Johnson, 1980). The assumption made here is that there exists a λ_0, such that $h(X,\lambda_o)$ is normally distributed with some mean μ and standard deviation σ. The subsequent algorithm demonstrates the technique of a parametric and nonparametric bootstrap method as well as the Box-Cox transformation:

1. From a sample $X_1,\ldots,X_n \sim f(x \mid \theta)$, estimate $g(\theta)$ using a statistic $\hat{g} = T(X_1,\ldots,X_n)$.
2. (Bootstrap Step) Generate B bootstrap samples, and compute $\hat{g}_1^*, \hat{g}_2^*,\ldots,\hat{g}_B^*$.
3. (Box-Cox Step) Find a $\hat{\lambda}$, such that $\hat{g}_1^{*\hat{\lambda}}, \hat{g}_2^{*\hat{\lambda}},\ldots,\hat{g}_B^{*\hat{\lambda}}$ is approximately normal.

The quantity $\hat{\lambda}$ is obtained by minimizing the mean square error for the linear fit of the power transformation

$$\frac{\hat{g}_{(1)}^{*\hat{\lambda}}-1}{\hat{\lambda}\dot{g}^{*\hat{\lambda}-1}}, \frac{\hat{g}_{(2)}^{*\hat{\lambda}}-1}{\hat{\lambda}\dot{g}^{*\hat{\lambda}-1}}, \ldots, \frac{\hat{g}_{(B)}^{*\hat{\lambda}}-1}{\hat{\lambda}\dot{g}^{*\hat{\lambda}-1}},$$

and

$$\Phi^{-1}\left(\frac{1}{B+1}\right), \Phi^{-1}\left(\frac{2}{B+1}\right), \ldots, \Phi^{-1}\left(\frac{B}{B+1}\right),$$

where $\hat{g}_{(i)}^*$ represents the i^{th} order statistic, $\dot{g}^*$ is the geometric mean, and

$\Phi^{-1}\left(\frac{i}{B+1}\right)$, $i = 1\ldots B$ denotes the standard normal quantiles. Using B+1 in the denominator is a simple and common technique used to avoid trying to find the standard normal quantile corresponding to p=1, which is undefined (∞). Well-known inference procedures for the normal case can then be used on the transformed $\hat{g}_i^*$, two examples of which are confidence intervals and hypothesis tests. Confidence intervals will be used within this thesis to measure the value of the Box-Cox transformation and bootstrap methods for one specific case. The confidence interval for the quantity $g^{\lambda_0}(\theta)$ is $\hat{g}^{\hat{\lambda}} \pm z_{\alpha/2} S^*\left(\hat{g}^{\hat{\lambda}}\right)$, where $S^*\left(\hat{g}^{\hat{\lambda}}\right)$ is the standard error of the bootstrap replicates of $\hat{g}^{\hat{\lambda}}$. The confidence interval for $g(\theta)$ is then the inverse transformation of the confidence interval. Because of the possibility that the lower endpoint of the confidence interval may be negative, there is the possibility of obtaining an undefined lower bound. In this case, the lower endpoint is set to 0.

CHAPTER III

THE BURR TYPE X DISTRIBUTION

The Burr type X distribution is a member of a family of twelve distributions introduced by Burr (1942), which were derived from the differential equation $dF(x)/dx = F(x)(1-F(x))g(x,F(x))$, where $g(x,y)$ is positive for the interval $0 \le y \le 1$, and $F(x)$ is the c.d.f. of the Burr family. The Burr XII is the most widely used, but within recent years the Burr X has received increasing attention. The two parameter Burr X, abbreviated as BurrX(θ,σ), has a c.d.f. given by

$$F(x \mid \theta,\sigma) = \left[1-e^{-(x/\sigma)^2}\right]^{\theta}, x>0, \theta>0, \sigma>0, \tag{3.1}$$

and the p.d.f. given by

$$f(x \mid \theta,\sigma) = \frac{2x\theta}{\sigma^2} e^{-(x/\sigma)^2}\left(1-e^{-(x/\sigma)^2}\right)^{\theta-1}, x>0, \theta>0, \sigma>0. \tag{3.2}$$

Ahmad, Fakhry, and Jaheen (1997) have studied the Burr type X distribution by looking at $R=P(Y<X)$. They found the m.l.e., the Bayes estimate, and the empirical Bayes estimate for R for the one-parameter Burr X (with no scale parameter σ). Surles and Padgett (1998) examined the two-parameter Burr X, with shape parameter θ and scale parameter σ given in (3.1) and (3.2). They considered inference for $R=P(Y<X)$ where X and Y are independent random variables that have a Burr X distribution with parameters (θ_1,σ_1) and (θ_2,σ_2), respectively. They examined two cases for which exact inference for R is possible. Case one is when the scale parameters σ_1 and σ_2 are assumed to both

be equal to one and case two is when σ_1 and σ_2 are equal but not necessarily one. They also describe approximate inference for R when $\sigma_1 \neq \sigma_2$.

Sartawi and Abu-Salih (1991) considered prediction intervals of the Bayesian type for order statistics of the one parameter Burr type X. Surles and Padgett (1998) also worked with the one-parameter Burr X and gave the uniform minimum variance unbiased estimator (UMVUE) for θ and R, and also discussed the distributional properties of the m.l.e. for θ.

Also associated with the Burr X distribution is the *exponentiated Weibull* distribution introduced by Mudholkar and Srivastava (1993), which is a generalization of the Weibull family in that there is an additional shape parameter. The Weibull Distribution was introduced by Weibull (1939) and is widely used in survival analysis. The c.d.f. of the exponentiated Weibull is given by

$$F(x \mid \alpha, \theta, \sigma) = \left[1 - e^{-(x/\sigma)^\alpha}\right]^\theta, x > 0, \theta > 0, \sigma > 0, \alpha > 0. \tag{3.3}$$

Setting $\alpha = 2$ the Burr type X distribution is obtained as a special case. If $\theta = 1$, then the Weibull distribution is obtained.

Inference for the shape parameter of the Burr X is obtained by finding the m.l.e. for the parameter θ as follows:

The likelihood function $L(\theta, \sigma)$ is given by $L(\theta, \sigma) = \prod_{i=1}^{n} f(x \mid \theta, \sigma)$, where f is given in (3.2),

$$L(\theta, \sigma) = \prod_{i=1}^{n} \frac{2\theta x_i}{\sigma^2} e^{-(x_i/\sigma)^2} \left[1 - e^{-(x_i/\sigma)^2}\right]^{\theta - 1}$$

$$= \frac{2^n \theta^n \prod_{i=1}^{n} x_i}{\sigma^{2n}} e^{-\sum_{i=1}^{n}(x_i/\sigma)^2} \prod_{i=1}^{n} \left[1 - e^{-(x_i/\sigma)^2}\right]^{\theta-1}, \tag{3.4}$$

where $x_1, x_2, \ldots, x_n$ is an observed random sample from the BurrX(θ, σ).

The log-likelihood function is then proportional to

$$\ell(\theta,\sigma) \propto n \ln\theta - 2n\ln\sigma - \sum_{i=1}^{n}(x_i/\sigma)^2 + (\theta-1)\sum_{i=1}^{n}\ln\left(1 - e^{-(x_i/\sigma)^2}\right). \tag{3.5}$$

To find the m.l.e. for θ, the partial derivative of (3.5) with respect to θ is found and set equal to 0. The resulting equation is then solved for θ,

$$\frac{\partial \ell}{\partial \theta} = \frac{n}{\theta} + \sum_{i=1}^{n} \ln\left(1 - e^{-(x_i/\sigma)^2}\right) \overset{set}{=} 0. \tag{3.6}$$

The m.l.e. for θ is thus

$$\hat{\theta} = \frac{n}{-\sum_{i=1}^{n} \ln\left(1 - e^{-(x_i/\hat{\sigma})^2}\right)}, \tag{3.7}$$

where $\hat{\sigma}$ is the m.l.e. for σ, found next.

To find the m.l.e. for σ (assuming σ is unknown), the partial derivative of (3.5) with respect to σ is found and set equal to 0. The resulting equation is then solved for σ,

$$\frac{\partial \ell}{\partial \sigma} = \frac{-2n}{\sigma} + \frac{2}{\sigma^3}\sum_{i=1}^{n} x_i^2 - \frac{2(\theta-1)}{\sigma^3}\sum_{i=1}^{n} \frac{x_i^2 e^{-(x_i/\sigma)^2}}{1 - e^{-(x_i/\sigma)^2}} = 0. \tag{3.8}$$

The form for $\hat{\theta}$ in (3.7) with σ instead of $\hat{\sigma}$ is substituted into (3.8) and after some minor simplification the following equation results,

$$-n+\frac{1}{\sigma^2}\sum_{i=1}^{n} x_i^2+\frac{\left[\left(n/\sum_{i=1}^{n}\ln\left(1-e^{-(x_i/\sigma)^2}\right)\right)+1\right]}{\sigma^2}\sum_{i=1}^{n}\frac{x_i^2 e^{-(x_i/\sigma)^2}}{1-e^{-(x_i/\sigma)^2}} \overset{set}{=} 0. \tag{3.9}$$

This equation has no closed-form solution, and numerical methods are required to solve it for σ.

To find inference for θ when σ is known, let $X \sim \text{BurrX}(\theta,\sigma)$, and define $Y = -\ln\left(1-e^{-(X/\sigma)^2}\right)$. The c.d.f. for Y is given by $F(y) = P(Y \le y)$, and some simple algebra yields the following:

$$\begin{aligned}
F(y) &= P\left(-\ln\left(1-e^{-(X/\sigma)^2}\right) \le y\right)\\
&= P\left(X \ge \sigma\sqrt{-\ln\left(1-e^{-y}\right)}\right)\\
&= 1-F\left(\sigma\sqrt{-\ln\left(1-e^{-y}\right)}\right), \text{ where F is given in (3.3)}\\
&= 1-\left[1-e^{-\left(\frac{\sigma\sqrt{\ln\left(1-e^{-y}\right)}}{\sigma}\right)^2}\right]^{\theta}\\
&= 1-e^{-\theta y}. \qquad (3.10)
\end{aligned}$$

F(y) given in (3.10) can be seen to be the c.d.f. of the exponential with mean $\frac{1}{\theta}$. Let $Y_i = -\ln\left(1-e^{-(X_i/\sigma)^2}\right)$, where $X_1, X_2, \ldots, X_n$ is a $\text{BurrX}(\theta,\sigma)$ random sample.

Then $Y_1, Y_2, \ldots, Y_n$ is an exponential random sample, and it follows that $\sum_{i=1}^{n} Y_i$ has a gamma distribution with parameters $\alpha = n$ and $\beta = \frac{1}{\theta}$. Since $\hat{\theta} = \frac{n}{\sum_{i=1}^{n} Y_i}$, as can be easily seen from (3.7), it follows that the distribution of $\hat{\theta}$ is given as inverted gamma, with parameters $\alpha = n$ and $\beta = \frac{1}{\hat{\theta}n}$. The p.d.f. of the inverted gamma is given as $f(x \mid \alpha, \beta) = \frac{1}{\Gamma(\alpha)\beta^{\alpha}} y^{-(\alpha+1)} e^{-\frac{1}{y\beta}}$ where $y > 0, \alpha > 0, \beta > 0$. The mean is $\frac{1}{(\alpha-1)\beta}$ and the variance is $\frac{1}{\beta^2(\alpha-1)^2(\alpha-2)}$. Thus, the mean of $\hat{\theta}$ is $\frac{n}{(n-1)}\theta$ and the variance is $\frac{\theta^2 n^2}{(n-1)^2(n-2)}$.

When σ is known, inference for θ can be illustrated by finding the confidence interval for θ. Hypothesis testing easily follows. Let $Z = \theta\sum_{i=1}^{n} -\ln\left[1 - e^{-(X_i/\sigma)^2}\right]$, for which we know the distribution of $\sum_{i=1}^{n} -\ln\left[1 - e^{-(X_i/\sigma)^2}\right]$ to be gamma$\left(n, \frac{1}{\theta}\right)$. Thus, the distribution of Z is gamma$(n,1)$, which is free of θ. Confidence intervals are computed using the pivot method as follows:

$$P\left(l_{\alpha/2} \le Z \le u_{\alpha/2}\right) = 1 - \alpha,$$

where $l_{\alpha/2}$ and $u_{\alpha/2}$ are appropriately defined gamma(n,1) critical values.

Thus, $P\left(l_{\alpha/2} \le \theta\sum_{i=1}^{n} -\ln\left(1 - e^{-(X_i/\sigma)^2}\right) \le u_{\alpha/2}\right) = 1 - \alpha$, which yields

$$P\left(\frac{l_{\alpha/2}}{\sum_{i=1}^{n} -\ln\left[1 - e^{-(X_i/\sigma)^2}\right]} \le \theta \le \frac{u_{\alpha/2}}{\sum_{i=1}^{n} -\ln\left[1 - e^{-(X_i/\sigma)^2}\right]}\right) = 1 - \alpha. \quad (3.11)$$

Therefore, $\left(\dfrac{l_{\alpha/2}}{\sum_{i=1}^{n} -\ln\left[1-e^{-(X_i/\sigma)^2}\right]}, \dfrac{u_{\alpha/2}}{\sum_{i=1}^{n} -\ln\left[1-e^{-(X_i/\sigma)^2}\right]} \right)$ is the $(1-\alpha)\times 100\%$ confidence interval for θ, which can also be seen from (3.7) to be $\left(\dfrac{\hat{\theta} l_{\alpha/2}}{n}, \dfrac{\hat{\theta} u_{\alpha/2}}{n} \right)$.

When σ is unknown, however, exact inference for θ cannot be obtained easily, because the sampling distribution for $\hat{\theta}$ is unknown. Surles (1999) show that the sampling distribution for $(\hat{\theta}, \hat{\sigma})$ is asymptotically normal, with variance-covariance matrix given by $I_n^{-1}(\theta, \sigma)$, where $I_n(\theta, \sigma)$ is the expected Fisher information matrix. Surles (1999) found the expected Fisher information matrix to be as follows:

$$I_n(\theta,\sigma) = \begin{pmatrix} I_{n;1,1} I_{n;1,2} \\ I_{n;1,2} I_{n;2,2} \end{pmatrix}, \tag{3.12}$$

where

$$I_{n;1,1} = -nE\left[\frac{\partial^2}{\partial\theta^2} \ln f(X \mid \theta,\sigma)\right] = \frac{n}{\theta^2}, \tag{3.13}$$

$$I_{n;1,2} = -nE\left[\frac{\partial^2}{\partial\theta\partial\sigma} \ln f(X \mid \theta,\sigma)\right] = \frac{2n\theta}{\sigma} \sum_{i=1}^{\infty} \frac{1}{i(\theta+i)(\theta+i-1)}, \tag{3.14}$$

and

$$I_{n;2,2} = -nE\left[\frac{\partial^2}{\partial\sigma^2} \ln f(X \mid \theta,\sigma)\right]$$

$$= \frac{(4-2\theta)n}{\sigma^2(\theta+1)} + \frac{2n\theta}{\sigma^2} \sum_{i=2}^{\infty} \left[\frac{3}{(\theta+i)(\theta+i-1)} + \frac{4(\theta-1)\sum_{j=1}^{i-1}\frac{1}{j}}{i(\theta+i-1)(\theta+i-2)} \right]. \tag{3.15}$$

where $f(X \mid \theta,\sigma)$ is given by (3.2). Thus, the asymptotic variance of $\hat{\theta}$ is given by $I^{-1}_{n;1,1} = \dfrac{I_{n;2,2}}{I_{n;1,1}I_{n;2,2} - I_{n;1,2}}$. A typical procedure is to assume that the distribution of $\hat{\theta}$ is exactly (or at least approximately) normal with variance that is estimated using $I^{-1}_{n;1,1}$. A confidence interval is thus computed as $\hat{\theta} \pm Z_{\alpha/2}\sqrt{\hat{I}^{-1}_{n;1,1}}$, where $\hat{I}^{-1}_{n;1,1}$ is $I^{-1}_{n;1,1}$ with θ and σ replaced by their m.l.e.'s. The effectiveness of this approximation will be studied and compared to the proposed Box-Cox bootstrap procedure described next to obtain approximate inference for θ.

Application of the algorithm described previously for the Box-Cox bootstrap procedure is as follows:

1. Let $X = X_1, X_2, \ldots, X_n \overset{iid}{\sim} \text{BurrX}(\theta,\sigma)$. Let $\hat{\theta}$ and $\hat{\sigma}$ be the m.l.e.'s for θ and σ computed using (3.7) and (3.9).
2. Generate B bootstrap samples $X^*_1, X^*_2, \ldots, X^*_B$, where $X^*_i = X^*_{i1}, X^*_{i2}, \ldots X^*_{in}$. The bootstrap sample X^*_i is generated by re-sampling from X (nonparametric bootstrap) or by generating X^*_i as a random sample from the $\text{BurrX}(\hat{\theta},\hat{\sigma})$ population (parametric bootstrap).
3. Let $\hat{\theta}^*_i$ be the i^{th} bootstrap replication of $\hat{\theta}$ computed using (3.7) and (3.9) with X^*_i.
4. Find $\hat{\lambda}$ such that $\hat{\theta}_1^{*\hat{\lambda}}, \ldots, \hat{\theta}_B^{*\hat{\lambda}}$ is approximately normal using the procedure

described previously.

5. Let s^* be the sample standard deviation of $\hat{\theta}_1^{*\hat{\lambda}}, \ldots, \hat{\theta}_B^{*\hat{\lambda}}$. Then the Box-Cox bootstrap confidence interval for θ is given as $\left(\hat{\theta}^{\hat{\lambda}} \pm Z_{\alpha/2} s^*\right)^{1/\hat{\lambda}}$.

In the next chapter, the Box-Cox bootstrap interval will be compared to the expected Fisher information matrix method, referred to as asymptotic variance method, for the shape parameter of the Burr X.

CHAPTER IV

SIMULATION STUDY

A simulation study was conducted to study the effectiveness of the proposed procedures for inference for the shape parameter θ from the Burr X. The factors that may have an effect on this inference are θ, σ, and n.

Within the simulation study, inference for θ will be carried out by generating N samples (the simulation size) and calculating the confidence interval for each. The number of times the true value of the parameter is within the interval is observed, and the coverage probability is estimated. The three methods of inference studied for θ will be:

1. Employing the asymptotic variance method described previously where the distribution of $\hat{\theta}$ is assumed to be (approximately) normal, regardless of sample size.
2. Employing the Box-Cox bootstrap method using a parametric bootstrap.
3. Employing the Box-Cox bootstrap method using a nonparametric bootstrap.

Simulations of this type were executed for all possible combinations of $\theta \in \{1,10,100\}$, $\sigma \in \{0.5,1,10\}$ and $n \in \{10,30,100,250\}$. The simulation size was constant at 1000. In the case of the Box-Cox bootstrap method, the value of B was 100.

For a specified θ, σ, and n:

1. Generate $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \overset{iid}{\sim} \text{BurrX}(\theta, \sigma)$, for which the vector $\boldsymbol{X}$ is easily generated using the probability integral transform $X = F^{-1}(U)$, where U is a continuous $U(0,1)$ random variable, and F is given in (3.1).
2. Compute $\hat{\theta}$ and $\hat{\sigma}$.
3. Use the asymptotic variance method described previously to compute the $(1-\alpha)\times 100\%$ confidence interval for θ, $\hat{\theta} \pm Z_{\alpha/2}\sqrt{\hat{I}_{n;1,1}^{-1}}$. This is denoted as the "A.V.M." method.
4. Use the nonparametric and parametric Box-Cox bootstrap methods, described previously, to compute $(1-\alpha)\times 100\%$ confidence intervals for θ denoted as the "Nonpar" and "Par" methods, respectively.

Note that simulations for each method were run independently, so the coverages given are independent. With all possible combinations of the above, 36 lines of data were acquired for each method. Tables 4.1-4.3 display the results of the simulation.

Results of the Simulation study

Table 4.1
Empirical Coverages for θ when $\sigma = 0.5$

θ , n		*A.V.M. /avglength*	*Nonpar/avglength*	*Par/avglength*
1	10	0.974 , 2.704298	0.783 , 9.419067	0.895 , 4.725
1	30	0.958 , 1.069028	0.894 , 1.217281	0.914 , 1.239
1	100	0.950 , 0.5348253	0.912 , 0.5519642	0.936 , 0.549
1	250	0.962 , 0.3288539	0.934 , 0.3315619	0.954 , 0.333
10	10	0.952 , 129.5771	0.764 , 25272.38	0.878 , 10110
10	30	0.963 , 21.83055	0.893 , 35.30616	0.921 , 31.43
10	100	0.958 , 9.411534	0.912 , 10.26940	0.926 , 10.28
10	250	0.958 , 5.546746	0.939 , 5.700736	0.934 , 5.719
100	10	0.787 , 2142.859*	0.791 , 2.958397e+10	0.888 , 1.437797e+10

100	30	0.932 , 616.2625	0.860 , 960.966	0.919 , 1461.036
100	100	0.954 , 183.9493	0.920 , 202.5919	0.924 , 201.14
100	250	0.961 , 99.84032	0.947 , 100.6508	0.946 , 101.09

Table 4.2
Empirical Coverages for θ when $\sigma = 1$

θ ,	n	*A.V.M./avlength*	*Nonpar/avglength*	*Par/avglength*
1	10	0.973 , 2.533277	0.780 , 10.16644	0.882 , 4.932
1	30	0.960 , 1.04566	0.894 , 1.257123	0.916 , 1.216
1	100	0.957 , 0.5303645	0.937 , 0.541378	0.920 , 0.557
1	250	0.958 , 0.329958	0.943 , 0.3313243	0.934 , 0.337
10	10	0.954 , 157.4266	0.779 , 72412.43	0.896 , 886.1
10	30	0.954 , 21.10889	0.910 , 32.30601	0.918 , 30.39
10	100	0.959 , 9.399558	0.933 , 10.07738	0.928 , 10.18
10	250	0.961 , 5.564317	0.935 , 5.728094	0.935 , 5.669
100	10	0.795 , 2553.427*	0.774, 2.679213e+10	0.887 , 9.18809e+10
100	30	0.936 , 673.7624	0.891 , 1265.985	0.923 , 1209
100	100	0.958 , 183.0238	0.908 , 205.5923	0.937 , 201.4
100	250	0.951 , 99.59474	0.936 , 100.7821	0.933 , 102.0

Table 4.3
Empirical Coverages for θ when $\sigma = 10$

θ ,	n	*A.V.M./avlength*	*Nonpar/avglength*	*Par/avglength*
1	10	0.971 , 2.654883	0.806 , 7.286623	0.868 , 8.199
1	30	0.962 , 1.053141	0.885 , 1.213119	0.913 , 1.207
1	100	0.958 , 0.5322846	0.945 , 0.5483497	0.934 , 0.555
1	250	0.957 , 0.3309815	0.943 , 0.33307	0.932 , 0.334
10	10	0.949 , 151.0562	0.761 , 279959.2	0.866 , 3046
10	30	0.945 , 21.28580	0.883 , 29.57917	0.925 , 30.76
10	100	0.954 , 9.246996	0.922 , 9.801392	0.935 , 10.06
10	250	0.957 , 5.58134	0.949 , 5.662918	0.938 , 5.767
100	10	0.780 , 2393.302*	0.802 , .468891e+12	0.901 , 1.4654e+10
100	30	0.925 , 590.7001	0.887 , 1533.191	0.911 , 1587
100	100	0.936 , 180.5987	0.920 , 200.7736	0.931 , 205.5
100	250	0.954 , 101.5455	0.940 , 100.0583	0.946 , 99.31

The entries within Tables 4.1-4.3 are the empirical coverages and the average lengths of the confidence intervals. There does not seem to be a relationship between θ and the empirical coverage. The size of the sample has a positive effect on the coverage probability (as expected) which is easily seen. The value of σ does not have any obvious effects on the empirical coverage of the methods. The low coverage probabilities occur for the A.V.M. method when $\theta = 100$ and $n = 10$ and this is attributed to $\hat{\theta}$ being very large on occasion. In this situation, the asymptotic variance is computed to be negative. This is most likely simply a computational problem. It is possible that when $\hat{\theta}$ is very large, the confidence interval would not cover θ anyway. If this is the case, then the estimated coverages given in Tables 4.1-4.3 would not have any significant bias. This cannot be taken for a given, however, and all cases where $n = 10$ are removed for the following analyses to have a balanced design.

An analysis of variance (ANOVA) was done to test the significance of the factors σ, θ and n, and to detect effects which were not immediately apparent, and can be seen in Tables 4.4-4.6. Only second-order interactions are included.

Table 4.4
ANOVA Table for A.V.M. data

Source	DF	SS	MS	F	P
theta	2	0.00088141	0.00044070	14.34	0.002
n	2	0.00039563	0.00019781	6.44	0.022
sigma	2	0.00016385	0.00008193	2.67	0.130
theta*n	4	0.00064415	0.00016104	5.24	0.023
theta*sigma	4	0.00015659	0.00003915	1.27	0.356
n*sigma	4	0.00006770	0.00001693	0.55	0.704
Error	8	0.00024585	0.00003073		
Total	26	0.00255519			

Table 4.5
ANOVA Table for Par data

Source	DF	SS	MS	F	P
theta	2	0.00001622	0.00000811	0.19	0.834
n	2	0.00206467	0.00103233	23.58	0.000
sigma	2	0.00005267	0.00002633	0.60	0.571
theta*n	4	0.00011644	0.00002911	0.66	0.634
theta*sigma	4	0.00022311	0.00005578	1.27	0.356
n*sigma	4	0.00017667	0.00004417	1.01	0.457
Error	8	0.00035022	0.00004378		
Total	26	0.00300000			

Table 4.6
ANOVA Table for Nonpar data

Source	DF	SS	MS	F	P
theta	2	0.0003961	0.0001980	1.40	0.301
n	2	0.0126650	0.0063325	44.78	0.000
sigma	2	0.0003672	0.0001836	1.30	0.325
theta*n	4	0.0003730	0.0000933	0.66	0.637
theta*sigma	4	0.0001535	0.0000384	0.27	0.888
n*sigma	4	0.0004726	0.0001181	0.84	0.539
Error	8	0.0011314	0.0001414		
Total	26	0.0155587			

Examining the ANOVA for the A.V.M. method it can be seen that, in addition to n, only θ has a significant effect, as well as the $\theta * n$ interaction, which was not apparent from Tables 4.1-4.3, and for the Nonpar and Par methods only n had a significant effect. It is also clear that the value of σ has little to no significance as seen by the value of the p-values for its main effects and any interactions it is involved in. This is consistent with simulations performed by Surles and Padgett (1998, 2001). Residual plots were produced, and there was no evidence to suggest that these results are not valid. There are some differences and similarities among the three techniques, and these are discussed in the following.

Except for large θ and small n, the A.V.M. method appears to provide conservative inference procedures for θ. That is, the coverages are, as a general rule, larger than the nominal 95%. Although caution must be taken in interpreting the coverages of the A.V.M. method when $\theta = 100$ and $n = 10$ because of the previously mentioned computational issues, it is apparent that the coverage would be below (probably well below) the nominal 95%. This can be seen by examining the coverages for $\theta = 100$ and $n = 30$. This behavior is inconsistent with what is seen when $\theta = 1$ and $\theta = 10$, and is likely the cause of the significant interaction seen in Table 4.4.

Unlike the A.V.M. method, the Nonpar and Par methods are consistently liberal. That is the coverages are consistently below the nominal 95%. In addition to this poor performance, the width of the confidence intervals are much larger than those for the A.V. M. method when n is small, and are even larger (albeit only slightly) for large n. With but a few exceptions, the performance of the Par method is, as expected, better than the performance of the Nonpar method. It is interesting to note, however, that the performance of the Nonpar and Par methods were not affected by the value of θ. This is significant because it represents an advantage over the A.V.M. method where the value of θ has a clear influence on the coverage probabilities. It would be of interest to study if the Box-Cox bootstrap method could be modified in some manner so that the coverage probabilities are improved while retaining this property.

If a researcher is to choose among the three methods for inference for the Burr X shape parameter, θ, it is clear that the choice would be the A.V.M. method. If, however, the researcher is limited to a small sample size and has reason to believe that θ is large, then neither of the three would likely provide satisfactory results. Even though the A.V.M. method is superior in this case of inference for the Burr X shape parameter θ, this does not discount the usefulness of the Box-Cox bootstrap method in other cases. Surles (1999) showed through a simulation study that inference for R=P(Y<X) (X and Y independent Burr X random variables) can be significantly improved when using the Box-Cox bootstrap method as compared to the method analogous to the A.V.M. method described here. If it would be possible to modify the Box-Cox bootstrap method to provide near-nominal coverages while remaining unaffected by the value of θ, then a researcher could use the Box-Cox bootstrap method with confidence regardless of the value of θ. Such an improvement is left as an open problem.

CHAPTER V

CONCLUSION

Within Chapter II, properties of bootstrapping and the Box-Cox transformation were given as well as the steps to obtaining a bootstrap sample, both nonparametrically and parametrically. Application of the Box-Cox transformation to the subsequent bootstrap sample was discussed. Also discussed was the asymptotic variance method to which the previous two methods were compared. Confidence intervals for each of the proposed methods were given.

In Chapter III, the Burr type X distribution was discussed. The p.d.f. and the c.d.f. of the Burr X were given. The m.l.e.'s for the shape parameter θ and scale parameter σ were given for the case when σ is both unknown and known. Exact inference for θ when σ is known was given. Also discussed was the asymptotic normality of the m.l.e.'s $\hat{\sigma}$ and $\hat{\theta}$, and the Fisher information matrix derived by Surles (1999) was reproduced. An algorithm for applying the Box-Cox bootstrap procedure in the Burr X case was also introduced.

In Chapter IV, the results of a simulation study to compare the coverage probabilities of the proposed methods were given. It was determined that the value of n had a significant effect upon the coverage probabilities, σ had little to no effect for all three methods. The value of θ had a significant effect on the coverages of the A.V.M. method while no such effect was observed for both Box-Cox bootstrap methods.

In conclusion, it has been shown that the Box-Cox bootstrap method is not an effective inference procedure for the shape parameter of the Burr type X distribution, whereas the asymptotic variance method utilizing the Fisher information matrix is a better inference procedure despite its conservative nature. As previously noted, however, the Box-Cox bootstrap method is an improvement over the A.V.M. method in some cases (Surles, 1999).

LIST OF REFERENCES

Ahmad, K.E., M.E. Fakhry, and Z.F. Jaheen, "Empirical Bayes Estimation of P(Y<X) and Characterization of Burr Type X Model," *Journal of Statistical Planning and Inference*, Vol. 64, pp. 297-308, 1997.

Box, G.E.P., and D.R. Cox, "An Analysis of Transformations," *Journal of the Royal Statistical Society B*, Vol. 26, pp. 211-252, 1964.

Burr, I.W., "Cumulative Frequency Functions," Annals *of Mathematical Statistics*, Vol. 13, pp. 215-222, 1942.

Chen, H., and W-Y. Loh, "Consistency of the Bootstrap For the Two-Sample T-Test," *Commununications in Statistics- Theory and Methods,* Vol. 20, pp. 997-1014, 1991, 1991.

Davison, A.C., and D.V. Hinkley, *Bootstrap Methods and their Applications, 2nd ed.*, Cambridge University Press, Cambridge United Kingdom, 1997.

Efron, B., "Bootstrap Methods: Another look at the jackknife," *Annals of Statistics*, Vol. 7 pp. 1-26, 1979.

Hernandez, F. and R.A. Johnson, "The Large-Sample Behavior of Transformations to Normality," *Journal of the American Statistical Association*, Vol. 75, Issue 372, pp. 855-861, 1980.

Logothetis, N., "Box-Cox Transformations and the Taguchi Method," *Applied Statistics*, Vol. 39, Issue I, pp. 31-48, 1990.

Mudholkar, G.S., and D.K. Srivastava, " Exponentiated Weibull family for analyzing bathtub failure-rate data," *IEEE Transactions on Reliability*, Vol. 42, pp. 299-302, 1993.

Sartawi, H.A., and M.S. Abu-Salih, "Bayesian Prediction Bounds for the Burr Type X Model," *Commununications in Statistics- Theory and Methods,* Vol. 20, pp. 2307-2330, 1991.

Surles, J.G., "The Scaled Burr Type X Distribution: Inferences and the Stress-Strength Problem," PhD Dissertation, Department of Statistics, University of South Carolina., 1999.

Surles, J.G. and W.J. Padgett, "Inference for P(Y<X) in the Burr Type X Model," *Journal of Applied Statistical Science*, Vol. 7, pp. 225-238, 1998.

Surles, J.G. and W.J. Padgett, "Inference for Reliability and Stress-Strength for a Scaled Burr Type X Distribution," *Lifetime Data Analysis*, Vol. 7, pp. 197-200, 2001.

Weibull, W., "A Statistical Theory of the Strength of Materials," *Ingeniors Vetenskaps Akademien Handlingar*, No. 153, 1939.

APPENDIX

SIMULATION STUDY

This appendix contains the R programs written to perform the simulations given in Chapter IV.

```
asyvar <- function(n, theta, sigma, tol=0.00001)
{
  # Compute the asymptotic variance for thetahat.
  i <- 1
  success <- FALSE
  sumb <- 0
  sumc <- 0
  sumj <- 0
  oldsumc <- 1000
  while (!success)
  {
    # Compute sums until successive steps are within tol.
    oldsumb <- sumb
    sumb <- sumb + 1/(i*(theta+i)*(theta+i-1))
    if (i >= 2)
    {
      sumj <- sumj + 1/(i-1)
      oldsumc <- sumc
      sumc <- sumc +
      3/((theta+i)*(theta+i-1))+(4*(theta-1)*sumj)/(i*(theta+i-1)*(theta+i-2))
    }
    i <- i+1
    if ((abs(oldsumb-sumb) <= tol) && (abs(oldsumc-sumc) <= tol))
    {
      success <- TRUE
    }
  }
  a <- n/theta**2
  b <- 2*n*theta/sigma*sumb
  c <- (4-2*theta)*n/(sigma**2*(theta+1))+2*n*theta/sigma**2*sumc
  ans <- c/(a*c-b**2)
  return(ans)
}

genburr <- function(n, theta, sigma)
{
  # Generage a BurrX random sample
  if (as.integer(n) != n || n <= 0)
  {
    stop ("ERROR: n must be a positive integer.\n")
  }
  if (theta <= 0)
  {
    stop ("ERROR: theta must be a positive real number.\n")
  }
  if (sigma <= 0)
  {
    stop ("ERROR: sigma must be a positive real number.\n")
  }
  u <- runif(n, min=0, max=1)
  x <- sigma*sqrt(-1*log(1-u**(1/theta)))
  # Make sure none of the x's are 0.  Happens from time-to-time.
  while (sum(x==0) > 0)
  {
```

```
    u <- runif(n, min=0, max=1)
    x <- sigma*sqrt(-1*log(1-u**(1/theta)))
  }
  return(x)
}

ftheta <- function(x, sigma)
{
  # Compute the MLE for theta given sigma.
  n <- length(x)
  temp <- x <= 0
  if (n == 0 || sum(temp) > 0)
  {
    stop ("ERROR: x must be a vector of positive real values.\n")
  }
  if (sigma <= 0)
  {
    stop ("ERROR: sigma must be a positive real value.\n")
  }
  n <- length(x)
  theta <- -1*n/sum(log(1-exp(-1*(x/sigma)**2)))
  return(theta)
}

fsigma <- function(sigma, x)
{
  # Function to solve to find the root for to find the MLE for
  # sigma.
  if (sigma <= 0)
  {
    stop ("ERROR: sigma must be a positive real number.\n")
  }
  n <- length(x)
  temp <- x <= 0
  if (n == 0 || sum(temp) > 0)
  {
    stop ("ERROR: x must be a vector of positive real values.\n")
  }
  sum1 <- sum(x**2)
  sum2 <- sum(x**2/(exp((x/sigma)**2)-1))
  ans <- -1*n+sum1/sigma**2-(ftheta(x,sigma)-1)/sigma**2*sum2
  return(ans)
}

mlesigma <- function(x, n)
{
  # Find the root of fsigma to find the MLE for sigma.
  # First, find lower and upper limits such that fsigma is finite.
  l <- 0.01
  success <- 0
  while ((success == 0) && (l < 50))
  {
    temp <- fsigma(l,x)
    if (is.finite(temp) && temp > 0)
    {
      success <- 1
    }
    else
    {
      l <- l + 0.01
```

```
    }
  }

  u <- 50
  success <- 0
  while ((success == 0) && (u > 0.01))
  {
    temp <- fsigma(u,x)
    if (is.finite(temp) && temp < 0)
    {
      success <- 1
    }
    else
    {
      u <- u - 0.01
    }
  }

  if ((1 >= u) || (1 >= 50) || (u <= 0.01))
  {
    return (-1)
  }
  else
  {
    # Now find the root of fsigma.
    soln <- uniroot (fsigma, c(1, u), tol=0.000001, x=x)
    return (soln$root)
  }
}

mletheta <- function(x, n, sigma)
{
  # This function is actually the same as ftheta, and
  # is redundant.
  thetahat <- -1*n/sum(log(1-exp(-1*(x/sigma)**2)))
  return (thetahat)
}

boxcox <- function(lambda, thetahatb, b, k2)
{
  # Find the value of lambda that provides the best
  # linear fit between thetahatb and standard normal
  # quantiles.
  k1 <- 1/(lambda*k2**(lambda-1))
  if (lambda!=0)
  {
    w <- k1*(thetahatb**lambda-1)
  }
  else
  {
    w <- k2*log(thetahatb)
  }
  w <- sort(w)
  p <- c(1:b)/(b+1)
  z <- qnorm(p)
  fit <- lm(w ~ z)
  sse <- sum(fit$residuals**2)
  return(sse)
}
```

```
param <- function(theta, sigma, alpha, n, b, simsize, skewcut=0)
{
  # Perform the parametric Box-Cox bootstrap simulation.
  library(e1071)
  thetahatb<- vector(mode="numeric",length=b)
  sigmahatb<- vector(mode="numeric",length=b)
  lower <- vector(mode="numeric", length=simsize)
  upper <- vector(mode="numeric", length=simsize)
  est <- vector(mode="numeric", length=simsize)
  lambdavec <- vector(mode="numeric", length=simsize)
  coverage <- 0
  intlen <- 0
  numzero <- 0
  starttime <- Sys.time()
  for (i in 1:simsize)
  {
    x <- genburr(n,theta,sigma)
    sigmahatx <- mlesigma(x, n)
    thetahatx <- mletheta(x, n, sigmahatx)
    j <- 1
    # Bootstrap step
    while (j <= b)
    {
      xb <- genburr(n,thetahatx,sigmahatx)
      sigmahatb[j] <- mlesigma(xb, n)
      if (sigmahatb[j] > 0)
      {
        thetahatb[j] <- mletheta(xb, n, sigmahatb[j])
        j <- j + 1
      }
    }
    est[i] <- thetahatx
    # Box-Cox step
    k2 <- sum(log(thetahatb))/b
    k2 <- exp(k2)
    skew <- skewness(thetahatb)
    if (abs(skew) <= skewcut)
    {
      lambdahat <- 1
    }
    else
    {
      lambdahat <- optimize(f=boxcox, interval=c(-10,10),
      thetahatb=thetahatb, b=b, k2=k2)
      lambdahat <- lambdahat$minimum
    }
    lambdavec[i] <- lambdahat
    if (lambdahat==0)
    {
     tthetahatb <- log(thetahatb)
     tthetahatx <- log(thetahatx)
    }
    else
    {
     tthetahatb <- thetahatb**lambdahat
     tthetahatx <- (thetahatx)**lambdahat
    }
    lowert <- tthetahatx - qnorm(1-alpha/2)*sd(tthetahatb)
    uppert <- tthetahatx + qnorm(1-alpha/2)*sd(tthetahatb)
    if (lambdahat == 0)
```

```
      {
        lower[i] <- exp(lowert)
        upper[i] <- exp(uppert)
      }
      else
      {
        lower[i] <- lowert ** (1/lambdahat)
        upper[i] <- uppert ** (1/lambdahat)
      }
      if (!is.finite(lower[i]))
      {
        lower[i] <- 0
      }
      if (!is.finite(upper[i]))
      {
        upper[i] <- 0
      }
      if (lower[i] > upper[i])
      {
        temp <- lower[i]
        lower[i] <- upper[i]
        upper[i] <- temp
      }
      intlen <- intlen + upper[i]   lower[i]
      if (lower[i] == 0)
      {
        numzero <- numzero + 1
      }
      cat(i, lower[i], theta, upper[i])
      if (lambdahat != 1)
      {
        cat (" TTT")
      }
      if ((theta >= lower[i]) && (theta <= upper[i]))
      {
        coverage <- coverage+1
      }
      else
      {
        cat (" ***")
      }
      runcover <- coverage / i
      cat (" Coverage=", runcover)
      currtime <- Sys.time()
      avgtime <- difftime(currtime, starttime, units="mins") / i
      remaintime <- avgtime * (simsize-i)
      cat (" approx", remaintime, "mins remaining\n")
    }
    coverage <- coverage/simsize
    intlen <- intlen / simsize
    ans <- list(interval=cbind(lower, est, upper, lambdavec),
    coverage=coverage, avgint=intlen, numzero=numzero)
    return (ans)
}

nonparam <- function(theta, sigma, alpha, n, b, simsize, skewcut=0)
{
    # Perform the nonparametric Box-Cox bootstrap simulation.
    library(e1071)
    thetahatb<- vector(mode="numeric",length=b)
```

```
sigmahatb<- vector(mode="numeric",length=b)
lower <- vector(mode="numeric", length=simsize)
upper <- vector(mode="numeric", length=simsize)
est <- vector(mode="numeric", length=simsize)
lambdavec <- vector(mode="numeric", length=simsize)
coverage <- 0
intlen <- 0
numzero <- 0
starttime <- Sys.time()
for (i in 1:simsize)
{
  x <- genburr(n,theta,sigma)
  sigmahatx <- mlesigma(x, n)
  thetahatx <- mletheta(x, n, sigmahatx)
  j <- 1
  # Bootstrap step
  while (j <= b)
  {
    xb <- sample(x, replace=TRUE)
    sigmahatb[j] <- mlesigma(xb, n)
    if (sigmahatb[j] > 0)
    {
      thetahatb[j] <- mletheta(xb, n, sigmahatb[j])
      j <- j + 1
    }
  }
  est[i] <- thetahatx
  # Box-Cox step
  k2 <- sum(log(thetahatb))/b
  k2 <- exp(k2)
  skew <- skewness(thetahatb)
  if (abs(skew) <= skewcut)
  {
    lambdahat <- 1
  }
  else
  {
    lambdahat <- optimize(f=boxcox, interval=c(-10,10),
    thetahatb=thetahatb, b=b, k2=k2)
    lambdahat <- lambdahat$minimum
  }
  lambdavec[i] <- lambdahat
  if (lambdahat==0)
  {
   tthetahatb <- log(thetahatb)
   tthetahatx <- log(thetahatx)
  }
  else
  {
   tthetahatb <- thetahatb**lambdahat
   tthetahatx <- (thetahatx)**lambdahat
  }
  lowert <- tthetahatx - qnorm(1-alpha/2)*sd(tthetahatb)
  uppert <- tthetahatx + qnorm(1-alpha/2)*sd(tthetahatb)
  if (lambdahat == 0)
  {
    lower[i] <- exp(lowert)
    upper[i] <- exp(uppert)
  }
  else
```

```
      {
        lower[i] <- lowert ** (1/lambdahat)
        upper[i] <- uppert ** (1/lambdahat)
      }
      if (!is.finite(lower[i]))
      {
        lower[i] <- 0
      }
      if (!is.finite(upper[i]))
      {
        upper[i] <- 0
      }
      if (lower[i] > upper[i])
      {
        temp <- lower[i]
        lower[i] <- upper[i]
        upper[i] <- temp
      }
      intlen <- intlen + upper[i]   lower[i]
      if (lower[i] == 0)
      {
        numzero <- numzero + 1
      }
      cat(i, lower[i], theta, upper[i])
      if (lambdahat != 1)
      {
        cat (" TTT")
      }
      if ((theta >= lower[i]) && (theta <= upper[i]))
      {
        coverage <- coverage+1
      }
      else
      {
        cat (" ***")
      }
      runcover <- coverage / i
      cat (" Coverage=", runcover)
      currtime <- Sys.time()
      avgtime <- difftime(currtime, starttime, units="mins") / i
      remaintime <- avgtime * (simsize-i)
      cat (" approx", remaintime, "mins remaining\n")
    }
    coverage <- coverage/simsize
    intlen <- intlen / simsize
    ans <- list(interval=cbind(lower, est, upper, lambdavec),
    coverage=coverage, avgint=intlen, numzero=numzero)
    return (ans)
}

classical <- function(theta, sigma, alpha, n, simsize)
{
    # Perform the A.V.M. simulation.
    library(e1071)
    lower <- vector(mode="numeric", length=simsize)
    upper <- vector(mode="numeric", length=simsize)
    est <- vector(mode="numeric", length=simsize)
    coverage <- 0
    intlen <- 0
    numzero <- 0
```

```
  starttime <- Sys.time()
  for (i in 1:simsize)
  {
    x <- genburr(n,theta,sigma)
    sigmahatx <- mlesigma(x, n)
    thetahatx <- mletheta(x, n, sigmahatx)
    j <- 1
    est[i] <- thetahatx

    stddev <- sqrt(asyvar(n, thetahatx, sigmahatx))
    if (is.finite(stddev))
    {
      lower[i] <- thetahatx   qnorm(1-alpha/2)*stddev
      upper[i] <- thetahatx + qnorm(1-alpha/2)*stddev
    }
    else
    {
      lower[i] <- 0
      upper[i] <- 0
      numzero <- numzero + 1
    }

    intlen <- intlen + upper[i]   lower[i]

    cat(i, lower[i], theta, upper[i])

    if ((theta >= lower[i]) && (theta <= upper[i]))
    {
      coverage <- coverage+1
    }
    else
    {
      cat (" ***")
    }
    runcover <- coverage / i
    cat (" Coverage=", runcover)
    currtime <- Sys.time()
    avgtime <- difftime(currtime, starttime, units="mins") / i
    remaintime <- avgtime * (simsize-i)
    cat (" approx", remaintime, "mins remaining\n")
  }
  coverage <- coverage/simsize
  intlen <- intlen / (simsize-numzero)
  ans <- list(interval=cbind(lower, est, upper), coverage=coverage,
  avgint=intlen, numzero=numzero)
  return (ans)
}
```

PERMISSION TO COPY

In presenting this thesis in partial fulfillment of the requirements for a master's degree at Texas Tech University or Texas Tech University Health Sciences Center, I agree that the Library and my major department shall make it freely available for research purposes. Permission to copy this thesis for scholarly purposes may be granted by the Director of the Library or my major professor. It is understood that any copying or publication of this thesis for financial gain shall not be allowed without my further written permission and that any user may be liable for copyright infringement.

Agree (Permission is granted.)

______________________________ ____________
Student Signature Date

Disagree (Permission is not granted.)

______________________________ ____________
Student Signature Date