

# Data dependent choice of the tuning parameter for the modified bootstrap

L. Santana (Hons. B.Sc.)

Mini-dissertation submitted in partial fulfilment  
of the requirements for the degree Magister Scientiae in Statistics at  
the Potchefstroom University for Christian Higher Education

Supervisor: Prof. J. W. H. Swanepoel

Co-supervisor: Prof. F.C. van Graan

2004

Potchefstroom

# Abstract

It is the purpose of this study to investigate a procedure for estimating the tuning parameter,  $m$ , in the modified bootstrap (also known as the  $m$ -out-of- $n$  bootstrap, the MOON bootstrap or the  $m/n$  bootstrap). The procedure which is to be investigated was first proposed by Bickel, Götze and van Zwet(1997). This estimation procedure was then further investigated theoretically by Bickel and Sakov(1999). In order to gain further insight, simulation studies were conducted using this data-based choice of  $m$ . The simulations involved constructing 90% confidence upper bounds through the use of the modified bootstrap method of percentile confidence upper bounds, known as the “Hybrid” and “Backwards” bounds. The simulation study provided, among other things, estimates for the coverage probabilities of the procedures. Interesting findings and conclusions of this study are reported.

# Uittreksel

Die doel van hierdie studie is om 'n beramingsmetode van die gladstryk parameter  $m$ , by die aanwending van die gewysigde skoensusmetode (ook bekend as die  $m$ -uit- $n$  skoensus, die MOON skoensus of die  $m/n$  skoensus), te ondersoek. Die prosedure wat deur Bickel, Götze en van Zwet (1997) voorgestel is, sal bestudeer word. Hierdie beramingsmetode is ook teoreties deur Bickel en Sakov (1999) ondersoek. Ten einde verdere inligting te bekom, is daar in hierdie studie Monte-Carlo simulasies gedoen deur van bogenaaamde data-gebaseerde keuse van  $m$  gebruik te maak. Die gewysigde skoensusmetode is aangewend om 90% persentiel vertrouens-bogrense vir beide die gemiddelde en variansie van 'n verdeling te konstrueer, wat ook bekend staan as "Kruising" en "Agteruit" bogrense. Die simulasie studies is hoofsaaklik daarop gemik om oordeckingswaarskynlikhede van die prosedures te beraam. Interessante bevindinge en gevolgtrekkings word bespreek.

# Summary

It is the purpose of this study to investigate a procedure for estimating the tuning parameter,  $m$ , in the modified bootstrap (also known as the  $m$ -out-of- $n$  bootstrap, the MOON bootstrap or the  $m/n$  bootstrap). The procedure which is to be investigated was first proposed by Bickel, Götze and van Zwet(1997). This estimation procedure was then further investigated theoretically by Bickel and Sakov(1999). In order to gain further insight, simulation studies were conducted using this data-based choice of  $m$ . The simulations involved constructing 90% confidence upper bounds through the use of the modified bootstrap method of percentile confidence upper bounds, known as the “Hybrid” and “Backwards” bounds. The simulation study provided, among other things, estimates for the coverage probabilities of the procedures.

Chapter 1 gives a broad outline of the non-parametric bootstrap and modified bootstrap methodologies, as well as a brief explanation as to how the two methods differ. Chapters 2 and 3 give more detailed descriptions of these two methods, while Chapter 4 provides an explanation of the methods followed to implement the procedures.

Chapter 2 deals with the non-parametric “classical” bootstrap procedure. It also explains the meaning of the more important aspects of the technique, namely the empirical distribution function, the double bootstrap, a bootstrap sample and the plug-in principle. It goes further to give details on the use of the bootstrap in estimating the standard error, bias, and confidence intervals, as well as the Monte-Carlo algorithm used to estimate these quantities in practice.

Chapter 3 concerns itself with the modified bootstrap. It focuses mainly on the construc-

tion of percentile confidence intervals and upper bounds using the “Hybrid” and “Backwards” percentile methods. In addition to this, it also describes the method of Bickel, Götze and van Zwet(1997) in more detail.

Chapter 4 describes the Monte-Carlo procedure used in the study. It explains the outputs, which can be found in Appendix A, as well as the algorithm followed by the simulation program, which can be found in Appendix B. Finally, it displays the findings reached by this study and the conclusions which can be inferred from them.

# Opsomming

Die doel van hierdie studie is om 'n beramingsmetode van die gladstryk parameter  $m$ , by die aanwending van die gewysigde skoelusmetode (ook bekend as die  $m$ -uit- $n$  skoelus, die MOON skoelus of die  $m/n$  skoelus), te ondersoek. Die prosedure wat deur Bickel, Götze en van Zwet (1997) voorgestel is, sal bestudeer word. Hierdie beramingsmetode is ook teoreties deur Bickel en Sakov (1999) ondersoek. Ten einde verdere inligting te bekom, is daar in hierdie studie Monte-Carlo simulاسies gedoen deur van bogenaaamde data-gebaseerde keuse van  $m$  gebruik te maak. Die gewysigde skoelusmetode is aangewend om 90% persentiel vertrouens-bogrense vir beide die gemiddelde en variانسie van 'n verdeling te konstrueer, wat ook bekend staan as "Kruising" en "Agteruit" bogrense. Die simulاسie studies is hoofsaaklik daarop gemik om oordekkingswaarskynlikhede van die prosedures te beraam.

Hoofstuk 1 gee 'n kort bespreking van die nie-parametriese klassieke skoelusmetode en die gewysigde skoelusmetode, asook 'n beknopte verduideliking van die verskille tussen hierdie metodes. Hoofstukke 2 en 3 bespreek die metodes meer breedvoerig, terwyl Hoofstuk 4 die praktiese implementering hiervan verduidelik.

Hoofstuk 2 bevat ook inligting oor belangrike begrippe soos die empiriese verdelingsfunksie, die dubbele skoelusmetode, 'n skoelussteekproef en die instop-beginsel. Verder word 'n breedvoerige verduideliking gegee van hoe die skoelusmetode aangewend kan word om die standaardfout en die sydigheid van 'n beramer te beraam, asook hoe vertrouensintervalle gekonstrueer kan word. 'n Monte-Carlo algoritme om hierdie groothede te benader in die praktyk word bespreek.

Die konstruksie van persentiel vertrouensintervalle en vertrouens-bogrense, deur gebruik-

making van die gewysigde “Kruising” en “Agteruit” skoelusmetode, word in Hoofstuk 3 gegee. Die beramingsmetode van Bickel, Götze en van Zwet (1997) word ook verduidelik.

In Hoofstuk 4 word die Monte-Carlo prosedure wat aangewend is, beskrywe. Aanhangsels A en B bevat die resultate van die simulاسies asook die rekenaar program. Die bevindinge en konklusies van hierdie studie word breedvoerig bespreek.

# Acknowledgements

The author wishes to express his gratitude towards:

Prof. J. W. H. Swanepoel, the promoter of this study, for his guidance, patience, and instruction.

My parents Manuel and Theresa Santana and my brothers, Sean and Peter, for their unwavering support throughout the years.

To God, without whom, nothing is possible.

# Notation

Symbol	Comment
$X_n$	A sample vector of size $n$ of independent, identically distributed random variables
$X_n^*$	A bootstrap sample vector of size $n$
$X_m^*$	A bootstrap sample vector of size $m$
$F$	The unknown distribution function of some random variable
$F_n$	The empirical distribution function
$\theta$	Some parameter of interest
$T_n(X_n; F)$ or $\hat{\theta}_n$	A random variable based on observations $X_n$ and unknown distribution function $F$
$T_n(X_n^*; F_n)$ or $\hat{\theta}_n^*$	A random variable based on observations $X_n^*$ and empirical distribution function $F_n$
$B$	The number of bootstrap replications
$P(A)$	The probability of some event $A$ calculated under $F$
$P^*(A)$	The probability of some event $A$ calculated under $F_n$
$Var_F(X)$ or $Var(X)$	The variance of some random variable $X$ calculated under $F$
$Var_{F_n}(X^*)$ or $Var_*(X^*)$	The variance of some random variable $X^*$ calculated under $F_n$
$E_F(X)$ or $E(X)$	The expected value of some random variable $X$ calculated under $F$
$E_{F_n}(X^*)$ or $E_*(X^*)$	The expected value of some random variable $X^*$ calculated under $F_n$
$n$	The size of the observed sample
$m$	The size of the modified bootstrap sample: $m \leq n$
$[x]$	Denotes the integer part of some value $x$

# Contents

1	An Introduction to the Bootstrap	1
1.1	The classical bootstrap	1
1.2	The modified bootstrap	2
2	The Classical Bootstrap	3
2.1	The plug-in principle	3
2.2	The bootstrap estimate of standard error	4
2.3	The bootstrap estimate of bias	5
2.4	Determining the accuracy of the standard error using the Double Bootstrap	6
2.5	Calculating Bootstrap Confidence Intervals	7
2.5.1	The Bootstrap-t Interval	8
2.5.2	The Percentile Method	9
2.5.3	The Bias-Corrected Percentile Method	9
2.5.4	The Accelerated Bias-Corrected Percentile Method	10
3	The Modified Bootstrap	12
3.1	Modified Bootstrap Confidence Bounds and Intervals	12
3.1.1	The “Hybrid” Percentile Method	13
3.1.2	The Backwards Percentile Method	16
3.2	Choosing the parameter $m$	19
4	Monte Carlo Study	21
4.1	The Algorithm	22
4.2	Results	25

# Chapter 1

## An Introduction to the Bootstrap

The bootstrap (introduced by Efron in 1979) is a non-parametric, computer intensive statistical methodology used in a wide range of statistical estimation applications. In fact, it could be argued that the bootstrap is applicable in nearly all circumstances requiring statistical estimation. One of the main uses of the bootstrap, however, is to describe variation in the data. That is, to estimate the standard error. In the following sections, established techniques of estimation using what is termed the classical or traditional bootstrap are investigated. This is then contrasted with the use of the 'm-out-of-n' or modified bootstrap. Firstly, however, it might be prudent to give a brief description of what, exactly, the classical bootstrap and the modified bootstrap entail.

### 1.1 The classical bootstrap

In classic statistical theory, model assumptions are usually always made in one way or another. A common assumption is that a given data set conforms to some or other statistical distribution. That is, one assumes that the data are generated from the assumed distribution in a random fashion. However, in practice it is the exception rather than the rule to be gifted with this sort of knowledge. Of course, large sample theory allows one to apply the central limit theorem and simply allow the data to come from a normal distribution. Unfortunately, this theory gives rise to some rather uncomfortable questions such as: "How large must the data set be before one can apply the central limit theorem?" and "If the sample is not large enough, what does one do then?"

The bootstrap has, in effect, given an answer to questions such as these. With the bootstrap one can estimate the distribution from which the data was generated. Estimation of this kind can be carried out on even small samples in which large sample theory could not hope to be applied. This makes analysis done using the bootstrap much more robust than methods which are based on some kind of assumption.

These incredible results are obtained by resampling the original sample by sampling with replacement and obtaining a large number (say,  $B$ ) of these resampled or bootstrap samples. For the classical bootstrap the sample sizes are the same as the original sample size. Some statistic is then calculated for each of these samples, these are then called bootstrap replications. These bootstrap replications are used to estimate the true distribution of the statistic. Once the distribution is known, all manner of calculations can be carried out on the statistic to reveal its properties.

## 1.2 The modified bootstrap

The modified bootstrap (also known as the 'm-out-of-n' bootstrap, MOON bootstrap or the  $m/n$  bootstrap) is an extension of the bootstrap and involves resampling a smaller number of elements from the original sample. This is different from the classical bootstrap which always samples the same number of elements as there are in the original sample. A more detailed description will be given in Chapter 3.

The reason for this modification is that it was found that in certain cases the classical bootstrap fails. It has subsequently been shown that these failures can be corrected through the use of this modified bootstrap (Swanepoel, 1986). Unfortunately, this method gives rise to a rather disturbing question: "How large should the modified sample sizes be?" That is, if  $n$  is the size of the original sample and  $m \leq n$  is the sample size of the bootstrap samples, what should  $m$  be? The choice of  $m = 2n/3$  has been shown to be "good" in a wide variety of applications but it is the hope of this study that there is a way of choosing  $m$  such that its value can be derived from the original sample. In other words, an attempt is being made to find a way of obtaining  $m$  data dependently.

# Chapter 2

## The Classical Bootstrap

Let  $\mathbf{X}_n = [X_1, X_2, \dots, X_n]'$  be a sample of independent, identically distributed random variables with some unknown distribution function  $F$ . One can estimate  $F$  using the empirical distribution function,  $F_n$ , which places mass  $\frac{1}{n}$  on each element of  $\mathbf{X}_n$ . Formally,  $F_n$  is defined as:

$$F_n(x) = \frac{1}{n} \sum_{i=1}^n I(X_i \leq x) ,$$

where

$$I(A) = \begin{cases} 1, & \text{if } A \text{ occurs} \\ 0, & \text{if } A^c \text{ occurs} \end{cases} \quad (2.1)$$

The empirical distribution can then be used to generate a bootstrap sample  $\mathbf{X}_n^* = [X_1^*, X_2^*, \dots, X_n^*]'$ , which is the same as sampling with replacement from the data  $X_1, X_2, \dots, X_n$ . In other words

$$P^*(X_j^* = X_i) = \frac{1}{n} \quad \forall i, j = 1, 2, \dots, n ,$$

where  $P^*$  is the probability calculated under  $F_n$ .

### 2.1 The plug-in principle

Let  $\theta$  be some parameter of interest. Now suppose  $\theta = \psi(F)$ , some functional of the unknown distribution function  $F$ , then the plug-in principle asserts that the bootstrap estimator of the parameter  $\theta$  is simply:

$$\hat{\theta}_n = \psi(F_n) ,$$

where  $F_n$  is the empirical distribution function.

This states that the parameter  $\theta$ , based on some functional of the unknown distribution function  $F$ , can be estimated by simply applying the same functional to the empirical distribution function  $F_n$ .

## 2.2 The bootstrap estimate of standard error

Given that  $\hat{\theta}_n = \hat{\theta}_n(X_1, X_2, \dots, X_n)$  estimates some unknown parameter  $\theta$ , then the estimate of the standard deviation can be estimated using the bootstrap.

Let  $\sigma(F)$  denote the standard deviation of  $\hat{\theta}_n$ . That is  $\sigma(F) = \sqrt{\text{Var}_F(\hat{\theta}_n)}$ . The plug-in principle allows one to estimate  $\sigma(F)$  by

$$\begin{aligned}\hat{\sigma}_n &= \sigma(F_n) \\ &= \sqrt{\text{Var}_{F_n}(\hat{\theta}_n^*)} \\ &= \sqrt{\text{Var}_*(\hat{\theta}_n^*)},\end{aligned}$$

where  $\hat{\theta}_n^* = \hat{\theta}_n(X_1^*, X_2^*, \dots, X_n^*)$  and are known as bootstrap replications.

The actual calculation of the bootstrap standard error of  $\hat{\theta}_n$  is computer intensive, but can be calculated no matter how complicated  $\hat{\theta}_n$  may be. To calculate the standard error, one can make use of a Monte-Carlo algorithm which proceeds as follows.

Monte-Carlo algorithm for approximating the standard error using the bootstrap:

1. Generate a sample  $X_1^*, X_2^*, \dots, X_n^*$  from the empirical distribution function,  $F_n$ , i.e. sample with replacement from  $X_1, X_2, \dots, X_n$ .
2. Calculate the statistic  $\hat{\theta}_n^* = \hat{\theta}_n(X_1^*, X_2^*, \dots, X_n^*) = \hat{\theta}_{n,1}^*$  for this first sample.
3. Independently repeat steps 1 and 2  $B$  times obtaining the bootstrap replications:

$$\hat{\theta}_{n,1}^*, \hat{\theta}_{n,2}^*, \dots, \hat{\theta}_{n,B}^*.$$

4. Now, calculate:

$$\hat{\sigma}_B = \sqrt{\frac{1}{B-1} \sum_{b=1}^B (\hat{\theta}_{n,b}^* - \hat{\theta}_{n,\cdot}^*)^2},$$

where

$$\hat{\theta}_{n,\cdot}^* = \frac{1}{B} \sum_{b=1}^B \hat{\theta}_{n,b}^*.$$

It is clear that  $\hat{\sigma}_B \rightarrow \hat{\sigma}_n = \sigma(F_n)$  as  $B \rightarrow \infty$  i.e. the Monte-Carlo bootstrap estimate of standard error approaches the theoretical bootstrap estimate of the standard error as the number of bootstrap repetitions increases to infinity.

## 2.3 The bootstrap estimate of bias

One can also use the bootstrap to estimate how far the expected value of the estimate deviates from the parameter value. This measure of discrepancy is also known as the bias.

If there exists some statistic,  $\hat{\theta}_n = \hat{\theta}_n(\mathbf{X}_n)$  based on the sample observations, then the bias is given by  $\beta(F) = E_F(\hat{\theta}_n(\mathbf{X}_n)) - \theta$ .

It is clear that through the use of the plug-in principle the bias can be estimated with the bootstrap. The bootstrap estimate for  $\beta(F)$  is

$$\begin{aligned} \hat{\beta}_n &= \beta(F_n) \\ &= E_{F_n}(\hat{\theta}_n^*) - \hat{\theta}_n \\ &= E_*(\hat{\theta}_n^*) - \hat{\theta}_n, \end{aligned}$$

where  $\hat{\theta}_n^* = \hat{\theta}_n(\mathbf{X}_n^*) = \hat{\theta}_n(X_1^*, X_2^*, \dots, X_n^*)$  and  $\hat{\theta}_n = \hat{\theta}_n(\mathbf{X}_n) = \hat{\theta}_n(X_1, X_2, \dots, X_n)$ .

Once again it is possible to evaluate the estimate  $\hat{\beta}_n$  using a Monte-Carlo algorithm.

Monte-Carlo algorithm for approximating the bias using the bootstrap:

1. Generate a sample  $X_1^*, X_2^*, \dots, X_n^*$  from the empirical distribution function,  $F_n$ , i.e. sample with replacement from  $X_1, X_2, \dots, X_n$ .
2. Calculate the statistic  $\hat{\theta}_n^* = \hat{\theta}_n(X_1^*, X_2^*, \dots, X_n^*) = \hat{\theta}_{n,1}^*$  for this first sample.

3. Independently repeat steps 1 and 2  $B$  times, obtaining the bootstrap replications:

$$\hat{\theta}_{n,1}^*, \hat{\theta}_{n,2}^*, \dots, \hat{\theta}_{n,B}^*.$$

4. Now, calculate:

$$\hat{\beta}_B = \hat{\theta}_{n,\cdot}^* - \hat{\theta}_n,$$

where

$$\hat{\theta}_{n,\cdot}^* = \frac{1}{B} \sum_{b=1}^B \hat{\theta}_{n,b}^*.$$

## 2.4 Determining the accuracy of the standard error using the Double Bootstrap

A useful aspect of the bootstrap estimates (of standard error or of bias) is that one is able to check their accuracy by re-applying the bootstrap to these estimates. This technique is known as the double bootstrap since it involves bootstrapping a bootstrap estimate. It is a logical extension of what has been discussed so far because the bootstrap can be applied to any statistical estimate and this naturally includes other bootstrap estimates.

If the bootstrap estimate of standard error of some estimate,  $\hat{\theta}_n$ , is given by  $\hat{\sigma}_n = \hat{\sigma}_n(X_1, X_2, \dots, X_n)$ , then the bootstrap estimate of the standard error of  $\hat{\sigma}_n$  is :

$$\hat{\hat{\sigma}}_n = \sqrt{\text{Var}_*(\hat{\sigma}_n(X_1^*, X_2^*, \dots, X_n^*))},$$

where  $\hat{\sigma}_n = \hat{\sigma}_n(X_1^*, X_2^*, \dots, X_n^*) = \sqrt{\text{Var}_{**}(\hat{\theta}_n(X_1^{**}, X_2^{**}, \dots, X_n^{**}))}$  and  $X_1^{**}, X_2^{**}, \dots, X_n^{**}$  is a bootstrap sample, sampled with replacement from  $X_1^*, X_2^*, \dots, X_n^*$  and  $\text{Var}_{**}$  denotes the variance with respect to  $\mathbf{X}_n^{**} = [X_1^{**}, X_2^{**}, \dots, X_n^{**}]'$  with  $\mathbf{X}_n^* = [X_1^*, X_2^*, \dots, X_n^*]'$  fixed.

The algorithm for calculating the estimate for the bootstrap estimate is as follows.

Monte-Carlo algorithm for approximating the standard error of the bootstrap standard error using the double bootstrap:

1. Generate a sample  $X_1^*, X_2^*, \dots, X_n^*$  from the empirical distribution function,  $F_n$ , i.e. sample with replacement from  $X_1, X_2, \dots, X_n$ .

(a) Generate  $X_1^{**}, X_2^{**} \dots X_n^{**}$  from  $X_1^*, X_2^*, \dots X_n^*$  and calculate  $\hat{\theta}_n(X_1^{**}, X_2^{**} \dots X_n^{**}) = \hat{\theta}_{n,1}^{**}$ .

(b) Repeat (a)  $B$  times to obtain  $\hat{\theta}_{n,1}^{**}, \hat{\theta}_{n,2}^{**}, \dots \hat{\theta}_{n,B}^{**}$ .

(c) Now calculate:

$$\hat{\sigma}_{n,1}^* = \sqrt{\frac{1}{B-1} \sum_{b=1}^B (\hat{\theta}_{n,b}^{**} - \hat{\theta}_{n,\cdot}^{**})^2},$$

where

$$\hat{\theta}_{n,\cdot}^{**} = \frac{1}{B} \sum_{b=1}^B \hat{\theta}_{n,b}^{**}.$$

2. Independently repeat step 1  $B$  times obtaining the bootstrap replications:  $\hat{\sigma}_{n,1}^*, \hat{\sigma}_{n,2}^*, \dots, \hat{\sigma}_{n,B}^*$ .

3. Now, calculate:

$$\hat{\sigma}_B = \sqrt{\frac{1}{B-1} \sum_{b=1}^B (\hat{\sigma}_{n,b}^* - \hat{\sigma}_{n,\cdot}^*)^2},$$

where

$$\hat{\sigma}_{n,\cdot}^* = \frac{1}{B} \sum_{b=1}^B \hat{\sigma}_{n,b}^*.$$

Note:  $\hat{\sigma}_B \rightarrow \hat{\sigma}_n$  as  $B \rightarrow \infty$ .

## 2.5 Calculating Bootstrap Confidence Intervals

In the following section the four different ways in which the bootstrap can be applied to construct confidence intervals for a parameter are investigated. The extension to confidence upper and lower bounds is arbitrarily easy and will not be discussed. The four methods are:

1. The Bootstrap-t Interval .
2. The Percentile Method.
3. The Bias-Corrected Percentile Method.

#### 4. The Accelerated Bias-Corrected Percentile Method.

The last three methods are all based on a percentile method. The algorithms used to calculate these all contain the same concept of arranging the bootstrap replicates in ascending order and then choosing the element that occurs at a certain index. The index is then calculated as some function of the number of bootstrap replications,  $B$ , and the chosen significance level  $\alpha$ . The first method, on the other hand, makes use of the more traditional concept of using a quantile from some distribution. In the first method the distribution, as well as its corresponding quantiles, are approximated at the specific level  $\alpha$ .

### 2.5.1 The Bootstrap-t Interval

Let  $\hat{\theta}_n^* = \hat{\theta}_n(X_1^*, X_2^*, \dots, X_n^*)$  be some bootstrap statistic, and let the number of bootstrap replications be  $B$ . Then  $\hat{\theta}_{n,b}^*$  for  $b = 1, 2, 3, \dots, B$  is the  $b^{\text{th}}$  bootstrap replication.

In order to calculate the interval, it is important to first "studentize" the statistic. This is accomplished in the following way: (The studentized statistic will be represented by  $Z^*$ )

$$Z_b^* = \frac{\hat{\theta}_{n,b}^* - \hat{\theta}_n}{\hat{\sigma}_{n,b}^*}, \quad b = 1, 2, \dots, B,$$

where  $\hat{\sigma}_{n,b}^*$  is the standard error of  $\hat{\theta}_{n,b}^*$  and  $\hat{\theta}_n = \hat{\theta}_n(X_1, X_2, \dots, X_n)$ . In other words, the statistic is centered around the parameter estimate  $\hat{\theta}_n$  and then divided by its own standard error,  $\hat{\sigma}_{n,b}^*$  (It may be necessary to use the bootstrap to calculate this value).

Next, find the value  $\hat{t}(\alpha)$  such that it satisfies the following:

$$\frac{1}{B} \sum_{b=1}^B I(Z_b^* \leq \hat{t}(\alpha)) = \alpha,$$

where  $I$  is defined as in equation (2.1).

Finally, the  $100(1 - \alpha)\%$  confidence interval for  $\theta$ , denoted  $I_t$ , can be approximated in the following way:

$$I_t = \left[ \hat{\theta}_n - \hat{t}\left(1 - \frac{\alpha}{2}\right) \cdot \hat{\sigma}_n; \hat{\theta}_n - \hat{t}\left(\frac{\alpha}{2}\right) \cdot \hat{\sigma}_n \right],$$

where  $\hat{\sigma}_n$  is the standard error of  $\hat{\theta}_n$  (possibly estimated using the bootstrap).

## 2.5.2 The Percentile Method

Let  $\hat{G}$  denote the distribution function of the bootstrap statistic  $\hat{\theta}_n^* = \hat{\theta}_n(X_1^*, X_2^*, \dots, X_n^*)$  i.e.  $\hat{G}(x) = P^*(\hat{\theta}_n^* \leq x)$ . The  $100(1 - \alpha)\%$  confidence interval for  $\theta$  is then:

$$I_p = \left[ \hat{G}^{-1} \left( \frac{\alpha}{2} \right); \hat{G}^{-1} \left( 1 - \frac{\alpha}{2} \right) \right] .$$

One can approximate this interval by making use of a Monte-Carlo algorithm:

Approximating the Percentile Confidence Interval using the Bootstrap

1. Generate  $X_1^*, X_2^*, \dots, X_n^*$  from the empirical distribution function,  $F_n$ , i.e. generate  $X_1^*, X_2^*, \dots, X_n^*$  by sampling with replacement from  $X_1, X_2, \dots, X_n$ .
2. Calculate  $\hat{\theta}_n^* = \hat{\theta}_n(X_1^*, X_2^*, \dots, X_n^*) = \hat{\theta}_{n,1}^*$ .
3. Repeat steps 1 and 2  $B$  times obtaining  $\hat{\theta}_{n,1}^*, \hat{\theta}_{n,2}^* \dots \hat{\theta}_{n,B}^*$ .
4. Obtain the order statistics  $\hat{\theta}_{n,(1)}^* \leq \hat{\theta}_{n,(2)}^* \leq \dots \leq \hat{\theta}_{n,(B)}^*$ .
5. The interval is then:

$$I_p(B) = \left[ \hat{\theta}_{n,(r)}^*; \hat{\theta}_{n,(s)}^* \right] ,$$

where

$$r = \left[ B \cdot \left( \frac{\alpha}{2} \right) \right] ,$$

and

$$s = \left[ B \cdot \left( 1 - \frac{\alpha}{2} \right) \right] .$$

## 2.5.3 The Bias-Corrected Percentile Method

Given that  $\hat{G}(x) = P^*(\hat{\theta}_n^* \leq x)$ , then the bias-corrected percentile  $100(1 - \alpha)\%$  confidence interval for  $\theta$  is given by:

$$I_{bc} = \left[ \hat{G}^{-1} \left\{ \Phi \left( 2z_0 - z \left( \frac{\alpha}{2} \right) \right) \right\}; \hat{G}^{-1} \left\{ \Phi \left( 2z_0 + z \left( \frac{\alpha}{2} \right) \right) \right\} \right] ,$$

where  $\Phi$  is the standard normal distribution function,  $\Phi \left( z \left( \frac{\alpha}{2} \right) \right) = 1 - \frac{\alpha}{2}$  and furthermore  $z_0 = \Phi^{-1}(\hat{G}(\hat{\theta}_n))$ .

(It is important to note that if  $\hat{G}(\hat{\theta}) = \frac{1}{2}$ , then  $z_0 = \Phi^{-1}(\hat{G}(\hat{\theta})) = 0$  and therefore  $I_{bc}$  reduces to  $I_p$ .)

To calculate an estimate for this interval make use of the same algorithm as in Section 2.5.2, except that the values of  $r$  and  $s$  become:

$$r = \left[ B \cdot \Phi \left( 2z_0 - z \left( \frac{\alpha}{2} \right) \right) \right]$$

and

$$s = \left[ B \cdot \Phi \left( 2z_0 + z \left( \frac{\alpha}{2} \right) \right) \right]$$

Also,  $\hat{G}(\hat{\theta}_n)$  appearing in the definition of  $z_0$ , can be approximated by:

$$\hat{G}(\hat{\theta}_n) = \frac{1}{B} \sum_{b=1}^B I(\hat{\theta}_{n,b}^* \leq \hat{\theta}_n).$$

#### 2.5.4 The Accelerated Bias-Corrected Percentile Method

Now, in addition to correcting for bias, it is also possible to correct for skewness. The accelerated bias-corrected percentile method does this by not only including the properties of the bias-corrected method but also by adjusting for any problems arising from skewness. The interval denoted  $I_{abc}$  is given by:

$$I_{abc} = \left[ \hat{G}^{-1} \left\{ \Phi \left( 2z_0 - b \left( \frac{\alpha}{2} \right) \right) \right\}; \hat{G}^{-1} \left\{ \Phi \left( 2z_0 + c \left( \frac{\alpha}{2} \right) \right) \right\} \right],$$

where

$$b \left( \frac{\alpha}{2} \right) = \frac{z \left( \frac{\alpha}{2} \right) - z_0}{1 - \hat{a} \cdot (z_0 - z \left( \frac{\alpha}{2} \right))} + z_0,$$

and

$$c \left( \frac{\alpha}{2} \right) = \frac{z \left( \frac{\alpha}{2} \right) + z_0}{1 - \hat{a} \cdot (z_0 + z \left( \frac{\alpha}{2} \right))} - z_0,$$

with

$$\hat{a} = \frac{1}{6} \cdot \frac{\sum_{i=1}^n U_i^3}{\left\{ \sum_{i=1}^n U_i^2 \right\}^{\frac{3}{2}}}.$$

The notation  $U_i$  is known as the jack-knife influence function of the original estimate  $\hat{\theta}_n = \hat{\theta}_n(X_1, X_2, \dots, X_n)$ , i.e.

$$U_i = (n-1)(\hat{\theta}_{n-1,[i]} - \hat{\theta}_{n-1,[i]}), \quad \forall i = 1, 2, \dots, n,$$

where  $\hat{\theta}_{n-1,[i]} = \hat{\theta}_{n-1}(X_1, X_2, \dots, X_{i-1}, X_{i+1}, \dots, X_n)$ , i.e.  $\hat{\theta}_{n-1,[i]}$  is calculated from the original sample data with the  $i^{\text{th}}$  element "deleted", and:

$$\hat{\theta}_{n-1,[i]} = \frac{1}{n} \sum_{i=1}^n \hat{\theta}_{n-1,[i]}.$$

$\Phi$ ,  $z_0$  and  $z\left(\frac{\alpha}{2}\right)$  have the same definitions as in the previous section.

(Note: If the measure of skewness,  $\hat{a}$ , is found to be zero then the interval  $I_{abc}$  reverts to the interval  $I_{bc}$ )

Once again the interval can be estimated using the Monte-Carlo algorithm described in the previous sections with the exception that the values of  $r$  and  $s$  be replaced by:

$$r = \left[ B \cdot \Phi \left( 2z_0 - b \left( \frac{\alpha}{2} \right) \right) \right] ,$$

and

$$s = \left[ B \cdot \Phi \left( 2z_0 + c \left( \frac{\alpha}{2} \right) \right) \right] .$$

Once again, the expression  $\hat{G}(\hat{\theta}_n)$  appearing in the definition of  $z_0$ , can be approximated by:

$$\hat{G}(\hat{\theta}_n) = \frac{1}{B} \sum_{b=1}^B \mathbb{I}(\hat{\theta}_{n,b}^* \leq \hat{\theta}_n) .$$

# Chapter 3

## The Modified Bootstrap

It can be shown that for the majority of statistics calculated in practice the bootstrap provides adequate distribution estimates, however, there are cases where the classical bootstrap fails to work. It was for this reason that the modified bootstrap method was created.

The modified bootstrap involves estimating the sampling distribution of some statistic,  $P_F(T_n(\mathbf{X}_n; F) \in A)$ , by the modified bootstrap estimate of this probability  $P^*(T_m(\mathbf{X}_m^*; F_n) \in A)$ ,  $m \leq n$ , where  $X_1^*, X_2^*, \dots, X_m^*$  is a bootstrap random sample, drawn with replacement from the original sample  $X_1, X_2, \dots, X_n$ , with  $m$  chosen such that  $m \rightarrow \infty$  as  $n \rightarrow \infty$  and  $\frac{m}{n} \rightarrow 0$  at some rate. It is clear that in the case where  $m = n$  the modified bootstrap reduces to the original, classical bootstrap.

Calculation of a Monte-Carlo estimate for the modified bootstrap distribution follows directly from the original bootstrap Monte-Carlo algorithm with the exception that the bootstrap samples are now of size  $m$  instead of  $n$  ( $m \leq n$ ).

### 3.1 Modified Bootstrap Confidence Bounds and Intervals

Let  $X_1, X_2, \dots, X_n$  be independent, identically distributed random variables from some unknown distribution function  $F$ . Let  $\hat{\theta}_n = \hat{\theta}_n(X_1, X_2, \dots, X_n)$  be an estimate for some parameter  $\theta$ . The following sections will now look at ways of creating  $100(1-\alpha)\%$  confidence bounds and intervals using the modified bootstrap procedure. This involves sampling  $X_1^*, X_2^*, \dots, X_m^*$

from the empirical distribution function  $F_n$  with  $(m \leq n)$ .

The following two methods will now be described:

1. The “Hybrid” Percentile Method.
2. The “Backwards” Percentile Method.

The first thing to be considered will be the construction of  $100(1 - \alpha)\%$  confidence upper bounds. The extension to  $100(1 - \alpha)\%$  confidence intervals follows similarly.

Suppose there exists some number  $c$  such that the following statement is valid:

$$P_F(\sqrt{n}(\hat{\theta}_n - \theta) \geq -c) = 1 - \alpha. \quad (3.1)$$

This implies that:

$$P_F(\theta \leq \hat{\theta}_n + \frac{c}{\sqrt{n}}) = 1 - \alpha.$$

Now if it is known what  $F$  is, then it would be easy to obtain a  $100(1 - \alpha)\%$  confidence upper bound for  $\theta$  through the expression:

$$I = \left( -\infty; \hat{\theta}_n + \frac{c}{\sqrt{n}} \right]. \quad (3.2)$$

Note: The value of  $c$  is obtained from the distribution function  $F$  i.e.  $c = c(F)$ . But, as has already been stated, the distribution function  $F$  is unknown. The value  $c$  is now estimated by making use of the modified bootstrap.

Using the modified bootstrap one can easily obtain an estimate for the probability expression in equation (3.1) using the (modified bootstrap) plug-in principle.

$$P^*(\sqrt{m}(\hat{\theta}_m^* - \hat{\theta}_n) \geq -\hat{c}) = 1 - \alpha \quad (3.3)$$

### 3.1.1 The “Hybrid” Percentile Method

#### “Hybrid” Percentile Method for Confidence Upper Bounds

The first method of constructing confidence bounds and intervals using the modified bootstrap is what is known as the “Hybrid” Percentile method. It involves methods similar to those found in Sections 2.5.2 to 2.5.4. Define:

$$\hat{G}(t) = P^*(\sqrt{m}(\hat{\theta}_m^* - \hat{\theta}_n) \leq t). \quad (3.4)$$

Using the above expression and equation (3.3) the following expression is obtained:

$$1 - P^*(\sqrt{m}(\hat{\theta}_m^* - \hat{\theta}_n) \leq -\hat{c}) = 1 - \alpha ,$$

or

$$1 - \hat{G}(-\hat{c}) = 1 - \alpha .$$

Therefore,

$$\hat{G}(-\hat{c}) = \alpha ,$$

i.e.

$$\hat{c} = -\hat{G}^{-1}(\alpha) .$$

This means that there is now an estimate for  $c$ . Now, using expression (3.2) and substituting in the value of  $\hat{c}$ :

$$\hat{I} = \left( -\infty; \hat{\theta}_n + \frac{\hat{c}}{\sqrt{n}} \right] ,$$

or

$$\hat{I} = \left( -\infty; \hat{\theta}_n - \frac{\hat{G}^{-1}(\alpha)}{\sqrt{n}} \right] . \quad (3.5)$$

The  $100(1 - \alpha)\%$  upper confidence bound  $\hat{I}$  can be easily estimated through the use of the following Monte-Carlo algorithm.

**Estimating the ‘‘Hybrid’’ Percentile Confidence Upper Bound using the Modified Bootstrap**

1. Draw  $X_1^*, X_2^*, \dots, X_m^*$  from  $F_n$  and calculate  $\sqrt{m}(\hat{\theta}_{m,1}^* - \hat{\theta}_n) = T_1^*$ , where  $\hat{\theta}_{m,1}^* = \hat{\theta}_m(X_1^*, X_2^*, \dots, X_m^*)$ .
2. Repeat the first step  $B$  times to obtain  $T_1^*, T_2^*, \dots, T_B^*$ .
3. Sort  $T_1^*, T_2^*, \dots, T_B^*$  from smallest to largest to obtain the order statistics  $T_{(1)}^* \leq T_{(2)}^* \leq \dots \leq T_{(B)}^*$ .
4. Now, approximate  $\hat{I}$  by:

$$\begin{aligned} \hat{I}_B &= \left( -\infty; \hat{\theta}_n - \frac{T_{([B \cdot \alpha])}^*}{\sqrt{n}} \right] \\ &= \left( -\infty; \hat{\theta}_n - \sqrt{\frac{m}{n}}(\hat{\theta}_{m,([B \cdot \alpha])}^* - \hat{\theta}_n) \right] , \end{aligned}$$

where  $\hat{\theta}_{m,(1)}^* \leq \hat{\theta}_{m,(2)}^* \leq \dots \leq \hat{\theta}_{m,(B)}^*$  are the order statistics of  $\hat{\theta}_{m,1}^*, \hat{\theta}_{m,2}^* \dots \hat{\theta}_{m,B}^*$ .

Note that  $\hat{I}_B \rightarrow \hat{I}$  as  $B \rightarrow \infty$ .

### “Hybrid” Percentile Method for Confidence Intervals

Extending the confidence upper bound to a confidence interval is relatively easy. Let there exist numbers  $a$  and  $b$  such that:

$$P_F(\sqrt{n}(\hat{\theta}_n - \theta) \geq -a) = 1 - \frac{\alpha}{2},$$

and

$$P_F(\sqrt{n}(\hat{\theta}_n - \theta) \leq -b) = 1 - \frac{\alpha}{2}.$$

Define:

$$G(t) = P_F(\sqrt{n}(\hat{\theta}_n - \theta) \leq t). \quad (3.6)$$

Hence,  $a = -G^{-1}(\alpha/2)$  and  $b = -G^{-1}(1 - \alpha/2)$ . Then a  $100(1 - \alpha)\%$  confidence interval for  $\theta$  is:

$$I = \left[ \hat{\theta}_n + \frac{b}{\sqrt{n}}; \hat{\theta}_n + \frac{a}{\sqrt{n}} \right],$$

or

$$I = \left[ \hat{\theta}_n - \frac{G^{-1}(1 - \frac{\alpha}{2})}{\sqrt{n}}; \hat{\theta}_n - \frac{G^{-1}(\frac{\alpha}{2})}{\sqrt{n}} \right].$$

It is possible to estimate this interval using the modified bootstrap by simply replacing  $G^{-1}$  with  $\hat{G}^{-1}$ . Therefore:

$$\hat{I} = \left[ \hat{\theta}_n - \frac{\hat{G}^{-1}(1 - \frac{\alpha}{2})}{\sqrt{n}}; \hat{\theta}_n - \frac{\hat{G}^{-1}(\frac{\alpha}{2})}{\sqrt{n}} \right],$$

where (as in equation (3.4))

$$\hat{G}(t) = P^*(\sqrt{m}(\hat{\theta}_m^* - \hat{\theta}_n) \leq t)$$

The same Monte-Carlo algorithm that was used for the “Hybrid” upper bound can be used here with the exception that the last step changes.

**Estimating the “Hybrid” Percentile Confidence Interval using the Modified Bootstrap**

1. Draw  $X_1^*, X_2^*, \dots, X_m^*$  from  $F_n$  and calculate  $\sqrt{m}(\hat{\theta}_{m,1}^* - \hat{\theta}_n) = T_1^*$ .

2. Repeat the first step  $B$  times to obtain  $T_1^*, T_2^*, \dots, T_B^*$ .
3. Sort  $T_1^*, T_2^*, \dots, T_B^*$  from smallest to largest to obtain the order statistics  $T_{(1)}^* \leq T_{(2)}^* \leq \dots \leq T_{(B)}^*$ .
4. Now, approximate  $\hat{I}$  by:

$$\begin{aligned} \hat{I}_B &= \left[ \hat{\theta}_n - \frac{T_{([B \cdot (1 - \frac{\alpha}{2})])}^*}{\sqrt{n}}; \hat{\theta}_n - \frac{T_{([B \cdot \frac{\alpha}{2}])}^*}{\sqrt{n}} \right] \\ &= \left[ \hat{\theta}_n - \sqrt{\frac{m}{n}} (\hat{\theta}_{m, ([B \cdot (1 - \frac{\alpha}{2})])}^* - \hat{\theta}_n); \hat{\theta}_n - \sqrt{\frac{m}{n}} (\hat{\theta}_{m, ([B \cdot \frac{\alpha}{2}])}^* - \hat{\theta}_n) \right], \end{aligned}$$

where  $\hat{\theta}_{m, (1)}^* \leq \hat{\theta}_{m, (2)}^* \leq \dots \leq \hat{\theta}_{m, (B)}^*$  are the order statistics of  $\hat{\theta}_{m, 1}^*, \hat{\theta}_{m, 2}^*, \dots, \hat{\theta}_{m, B}^*$ .

Note that  $\hat{I}_B \rightarrow \hat{I}$  as  $B \rightarrow \infty$ .

### 3.1.2 The Backwards Percentile Method

#### “Backwards” Percentile Method for Confidence Upper Bounds

The “Backwards” Percentile Method of confidence bound and interval approximation involves using the assumption that the distribution of  $\sqrt{n}(\hat{\theta}_n - \theta)$  is symmetric about zero, i.e. the distribution of  $\sqrt{n}(\hat{\theta}_n - \theta)$  is equivalent to the distribution of  $\sqrt{n}(\theta - \hat{\theta}_n)$  or

$$\mathcal{D}(\sqrt{n}(\hat{\theta}_n - \theta)) = \mathcal{D}(\sqrt{n}(\theta - \hat{\theta}_n)).$$

Using this assumption, rewrite equation (3.1) as:

$$P_F(\sqrt{n}(\theta - \hat{\theta}_n) \geq -c) = 1 - \alpha. \quad (3.7)$$

Now, rewriting the above equation the following is obtained:

$$P_F(\sqrt{n}(\hat{\theta}_n - \theta) \leq c) = 1 - \alpha. \quad (3.8)$$

It is now possible to estimate this probability using the modified bootstrap:

$$P^*(\sqrt{m}(\hat{\theta}_m^* - \hat{\theta}_n) \leq \hat{c}) = 1 - \alpha, \quad (3.9)$$

or equivalently:

$$\hat{G}(\hat{c}) = 1 - \alpha,$$

where  $\hat{G}$  is given by equation (3.4).

Therefore, the estimate for the interval in equation (3.2) is:

$$\hat{I} = \left( -\infty; \hat{\theta}_n + \frac{\hat{G}^{-1}(1-\alpha)}{\sqrt{n}} \right]$$

One can estimate  $\hat{I}$  using the following Monte-Carlo algorithm.

Estimating the “Backwards” Percentile Confidence Upper Bound using the Modified Bootstrap

1. Draw  $X_1^*, X_2^*, \dots, X_m^*$  from  $F_n$  and calculate  $\sqrt{m}(\hat{\theta}_{m,1}^* - \hat{\theta}_n) = T_1^*$ .
2. Repeat the first step  $B$  times to obtain  $T_1^*, T_2^*, \dots, T_B^*$ .
3. Sort  $T_1^*, T_2^*, \dots, T_B^*$  from smallest to largest to obtain the order statistics  $T_{(1)}^* \leq T_{(2)}^* \leq \dots \leq T_{(B)}^*$ .
4. Now, approximate  $\hat{I}$  by:

$$\begin{aligned} \hat{I}_B &= \left( -\infty; \hat{\theta}_n + \frac{T_{([B \cdot (1-\alpha)])}^*}{\sqrt{n}} \right] \\ &= \left( -\infty; \hat{\theta}_n + \sqrt{\frac{m}{n}} (\hat{\theta}_{m,([B \cdot (1-\alpha)])}^* - \hat{\theta}_n) \right], \end{aligned}$$

where  $\hat{\theta}_{m,(1)}^* \leq \hat{\theta}_{m,(2)}^* \leq \dots \leq \hat{\theta}_{m,(B)}^*$  are the order statistics of  $\hat{\theta}_{m,1}^*, \hat{\theta}_{m,2}^* \dots \hat{\theta}_{m,B}^*$ .

Note that  $\hat{I}_B \rightarrow \hat{I}$  as  $B \rightarrow \infty$ .

### “Backwards” Percentile Method for Confidence Intervals

To construct  $100(1-\alpha)\%$  “Backwards” confidence intervals, which is a simple extension of the confidence bound, the numbers  $a$  and  $b$  need to be found that satisfy the expressions:

$$P_F(\sqrt{n}(\hat{\theta}_n - \theta) \leq a) = 1 - \frac{\alpha}{2},$$

and

$$\begin{aligned} P_F(\sqrt{n}(\hat{\theta}_n - \theta) \geq b) &= 1 - P_F(\sqrt{n}(\hat{\theta}_n - \theta) \leq b) \\ &= 1 - \frac{\alpha}{2}. \end{aligned}$$

Therefore

$$P_F(\sqrt{n}(\hat{\theta}_n - \theta) \leq b) = \frac{\alpha}{2}.$$

This leads to the following expressions for  $a$  and  $b$ :

$$a = G^{-1}(1 - \alpha/2),$$

and

$$b = G^{-1}(\alpha/2),$$

where  $G$  is given by equation (3.6). Then a  $100(1 - \alpha)\%$  confidence interval for  $\theta$  is:

$$I = \left[ \hat{\theta}_n + \frac{b}{\sqrt{n}}; \hat{\theta}_n + \frac{a}{\sqrt{n}} \right],$$

or

$$I = \left[ \hat{\theta}_n + \frac{G^{-1}(\frac{\alpha}{2})}{\sqrt{n}}; \hat{\theta}_n + \frac{G^{-1}(1 - \frac{\alpha}{2})}{\sqrt{n}} \right].$$

The estimate for the interval  $I$  can be found by using the modified bootstrap, by substituting  $G$  with its modified bootstrap estimate  $\hat{G}$ .

$$\hat{I} = \left[ \hat{\theta}_n + \frac{\hat{G}^{-1}(\frac{\alpha}{2})}{\sqrt{n}}; \hat{\theta}_n + \frac{\hat{G}^{-1}(1 - \frac{\alpha}{2})}{\sqrt{n}} \right],$$

where  $\hat{G}$  is given by equation (3.4).

Once again, the interval  $\hat{I}$  can be estimated by an application of a Monte-Carlo algorithm: Estimating the ‘‘Backwards’’ Percentile Confidence Interval using the Modified Bootstrap

1. Draw  $X_1^*, X_2^*, \dots, X_m^*$  from  $F_n$  and calculate  $\sqrt{m}(\hat{\theta}_{m,1}^* - \hat{\theta}_n) = T_1^*$ .
2. Repeat the first step  $B$  times to obtain  $T_1^*, T_2^*, \dots, T_B^*$ .
3. Sort  $T_1^*, T_2^*, \dots, T_B^*$  from smallest to largest to obtain the order statistics  $T_{(1)}^* \leq T_{(2)}^* \leq \dots \leq T_{(B)}^*$ .
4. Now, approximate  $\hat{I}$  by:

$$\begin{aligned} \hat{I}_B &= \left[ \hat{\theta}_n + \frac{T_{([B \cdot (\frac{\alpha}{2})])}^*}{\sqrt{n}}; \hat{\theta}_n + \frac{T_{([B \cdot (1 - \frac{\alpha}{2})])}^*}{\sqrt{n}} \right] \\ &= \left[ \hat{\theta}_n + \sqrt{\frac{m}{n}}(\hat{\theta}_{m,([B \cdot (\frac{\alpha}{2})])}^* - \hat{\theta}_n); \hat{\theta}_n + \sqrt{\frac{m}{n}}(\hat{\theta}_{m,([B \cdot (1 - \frac{\alpha}{2})])}^* - \hat{\theta}_n) \right], \end{aligned}$$

where  $\hat{\theta}_{m,(1)}^* \leq \hat{\theta}_{m,(2)}^* \leq \dots \leq \hat{\theta}_{m,(B)}^*$  are the order statistics of  $\hat{\theta}_{m,1}^*, \hat{\theta}_{m,2}^* \dots \hat{\theta}_{m,B}^*$ .

Once again, it can be seen that  $\hat{I}_B \rightarrow \hat{I}$  as  $B \rightarrow \infty$ .

## 3.2 Choosing the parameter $m$

Up until now the choice of  $m$  has always depended on some sort of heuristic or rule of thumb. While the performance of these methods have fared fairly well in certain applications, they have also been observed to quite poorly in a variety of others. This discrepancy from application to application makes one wish that a single “rule” existed that could encompass all manner of applications. Considering the problem from this view point it seems obvious that a more scientific method should be prescribed for selecting the parameter  $m$ . A logical choice would be an  $m$  chosen data-dependently. It is unfortunate that this problem has received very little attention in the literature.

Bickel, Götze, and van Zwet (1997) suggested a way to choose  $m$  data-dependently. Their basic idea was to choose the parameter  $m$  in such a way that it would minimise a specific criteria. In this case, the criteria is the absolute difference between a modified bootstrap statistic calculated using the parameter equal to some value  $m$ , and the same statistic calculated using  $m/2$ . The value  $m$  that minimises the criteria is believed to be the optimal choice of  $m$ . This suggestion was further considered theoretically (as  $n \rightarrow \infty$ ) by Götze and Rackauskas (1999).

The method of selecting  $m$  data-dependently suggested by Bickel, Götze, and van Zwet can be applied to the problem of confidence upper bounds (see Section 3.1) in the following way:

### Case 1 (“Hybrid” Percentile Confidence Upper Bound)

Find a value of  $m$ , say  $\hat{m}$ , that minimizes the function:

$$A_{m1} = \left| \sqrt{m} \left\{ \hat{\theta}_{m,([B \cdot \alpha])}^* - \hat{\theta}_n \right\} - \sqrt{\left\lfloor \frac{m}{2} \right\rfloor} \left\{ \hat{\theta}_{\left\lfloor \frac{m}{2} \right\rfloor, ([B \cdot \alpha])}^* - \hat{\theta}_n \right\} \right|. \quad (3.10)$$

In other words,  $\hat{m} = \arg \min_m A_{m1}$ .

## Case 2 (“Backwards” Percentile Confidence Upper Bound)

Find a value of  $m$ , say  $\hat{m}$ , that minimizes the function:

$$A_{m2} = \left| \sqrt{m} \left\{ \hat{\theta}_{m, ([B \cdot (1-\alpha)])}^* - \hat{\theta}_n \right\} - \sqrt{\left[ \frac{m}{2} \right]} \left\{ \hat{\theta}_{\left[ \frac{m}{2} \right], ([B \cdot (1-\alpha)])}^* - \hat{\theta}_n \right\} \right|. \quad (3.11)$$

In other words,  $\hat{m} = \arg \min_m A_{m2}$ .

The question that arises is whether this data-dependent choice of  $m$  has any practical use for typical sample sizes and data.

In the following chapter the performance of  $\hat{m}$  will be investigated by means of a limited Monte-Carlo study, restricting our attention to confidence bounds.

# Chapter 4

## Monte Carlo Study

This chapter will describe the simulation study and report the results obtained. The simulation was executed with the intention of finding coverage probabilities of the confidence upper bounds described in Section 3.1. The data-based choice of the parameter  $m$  used in these methods was calculated using the methods found in Section 3.2. Finally, these results were contrasted with results obtained from calculating the upper bound using the classical bootstrap ( $m = n$ ), and for two other fixed choices of  $m$ , namely  $m = n/3$  and  $m = 2n/3$ .

The following is a brief outline of the steps involved in the algorithm, this will be followed by a more detailed description. The basic algorithm involves determining the parameter  $m$  data dependently, this calculation is described in Section 3.2 and will not be repeated here. The calculated  $m$ , which will be denoted by  $\hat{m}$ , is then used in the calculation of an upper bound using a modified bootstrap procedure, such as those discussed in Chapter 3, Section 3.1. Implemented in this application of the algorithm were the two methods of calculating confidence upper bounds, they are the Hybrid and Backwards methods. Each of these two methods has its own unique procedure of calculating both the value  $\hat{m}$ , and the confidence upper bound. These calculations are then carried out on various samples of differing sizes and sampling distributions.

The second part of the algorithm calculates the same confidence upper bounds as those found in the first, data-dependent part except that it makes use of a fixed  $m$ . The values of  $m$  were chosen to be  $n/3$ ,  $2n/3$ , and  $n$  where  $n$  is the original sample size. These  $m$ 's were then used in the Modified Bootstrap calculations of the upper bounds.

These confidence upper bounds were then calculated for both the mean and the variance.

The output of all these procedures will be compared with one another and conclusions drawn from them.

The program applied can be found in Appendix B. It is written in Visual Fortran version 6.6.

## 4.1 The Algorithm

The program applied to this problem makes use of several parameters in its calculations.

These parameters are the following:

1. The number of Monte-Carlo repetitions, MC. This was set to a fairly large number, 10000, in order to reduce the amount of variation exhibited in the final results.
2. The number of bootstrap repetitions, B. This was also set to a large number, 1000, in order to increase the accuracy of individual bootstrap estimates. In the descriptions that follow, one will find many bootstrap calculations, all of which share the same number of bootstrap repetitions.
3. The significance level,  $\alpha$ . The significance level of the confidence upper bounds was chosen to be 0.1. Thus, all of the following calculations are attempting to create 90 % confidence upper bounds.
4. The sample sizes,  $n$ . The sample sizes were chosen to be relatively small in the belief that the modified bootstrap would still provide satisfactory results. For the calculation of the confidence upper bound of the mean, the sample sizes chosen were:  $n = 20, 50, 100$  and  $200$ . When calculating the confidence upper bound of the variance, the sample sizes were increased to:  $n = 50, 100, 200$  and  $300$ . The sample sizes were increased because it was felt that the variance is a more 'difficult' parameter to estimate accurately and therefore required larger samples.

The simulation results were obtained by the following algorithm:

1. Generate data from one of the following distributions: Exponential distribution with  $\lambda = 1$ , the F(5,8) distribution, the Weibull distribution with shape parameter 0.5

and scale parameter 1, the Standard Normal distribution  $N(0,1)$  contaminated with a  $N(1,0.01)$  distribution with probability 0.25 and Standard Normal distribution  $N(0,1)$  contaminated with a  $N(1,0.01)$  distribution with probability 0.5. This step yields the sample data set  $X_1, X_2, \dots, X_n$ .

2. Calculate the true value of the parameter of interest, which is denoted by  $\theta$ . The parameters which will be calculated are the mean or expected value and the variance.

(a) Exponential distribution:

The expected value is given by:

$$E(X) = 1/\lambda .$$

The variance is given by:

$$\text{Var}(X) = 1/\lambda^2 .$$

Thus, with  $\lambda = 1$ , both the expected value and the variance are equal to 1.

(b) The  $F(m,n)$  distribution:

The expected value is given by:

$$E(X) = n/(n - 2) , (n > 2) .$$

The variance is given by:

$$\text{Var}(X) = \frac{2n^2(m + n - 2)}{m(n - 4)(n - 2)^2} , (n > 4) .$$

Thus with  $m = 5$  and  $n = 8$ , the expected value is  $1\frac{1}{3}$  and the variance is 1.956.

(c) The Weibull distribution with shape parameter  $\gamma$  and scale parameter  $c$ .

The expected value is given by:

$$E(X) = \frac{\Gamma(1 + \frac{1}{\gamma})}{c^{1/\gamma}} .$$

The variance is given by:

$$\text{Var}(X) = \frac{\Gamma(1 + \frac{2}{\gamma})}{c^{2/\gamma}} - \left[ \frac{\Gamma(1 + \frac{1}{\gamma})}{c^{1/\gamma}} \right]^2 .$$

Thus with  $c = 1$  and  $\gamma = 0.5$  it is found that the expected value is 2 and the variance is 20.

- (d) The contaminated Normal distribution: ( i.e. A  $N(\mu_1, \sigma_1^2)$  distribution contaminated with a  $N(\mu_2, \sigma_2^2)$  distribution with probability  $p$ ):

The expected value is given by:

$$E(X) = (1 - p)\mu_1 + p\mu_2 .$$

The variance is given by:

$$\text{Var}(X) = (1 - p)\sigma_1^2 + p\sigma_2^2 + p(1 - p)(\mu_1 - \mu_2)^2 .$$

Thus for  $\mu_1 = 0$ ,  $\mu_2 = 1$ ,  $\sigma_1^2 = 1$ ,  $\sigma_2^2 = 0.01$  and  $p = 0.25$  it can be seen that the expected value is 0.25, and the variance is 0.94. For  $p = 0.5$  the expected value is 0.5, and the variance is 0.755.

3. This step in the algorithm determines the choice of  $m$  to be used in the calculation of the modified bootstrap confidence upper bounds. The parameter  $m$  can be chosen in one of two ways:
  - (a) Choose  $m$  arbitrarily. For this case  $m$  was chosen to be either one third of the sample size  $n$ , two thirds of the sample size  $n$  or equal to  $n$  (classical bootstrap method), i.e.  $m = n/3$ ,  $m = 2n/3$  or  $m = n$ .
  - (b) Choose  $m$  data-dependently by making use of the procedure outlined in Section 3.2. This procedure makes use of the bootstrap sample data in order to choose  $m$ , the resulting value chosen will be denoted by  $\hat{m}$ . As has already seen in Section 3.2, the method used to calculate  $\hat{m}$  is determined by whether one wants to calculate the Hybrid or the Backwards percentile confidence upper bound.
4. The confidence upper bounds are now calculated using one of the methods found in Section 3.1. In this step the choice of calculating either the "Hybrid" or "Backwards" confidence upper bound must correspond with the choice made in step 3(b).
5. The confidence upper bound value is then compared to the actual parameter calculated in step 2. If the upper bound value is greater than the actual parameter value then it is marked as a 'success', otherwise it is marked as a 'failure'. For simplicity, a success is denoted by a "1" and a failure by a "0".

6. Repeat steps 1 to 5 MC times. Count the number of times that the confidence upper bound is found to successfully cover the parameter, and then divide that total by the number of Monte-Carlo repetitions, MC. The resulting answer gives an approximation of the coverage probability of the particular method employed. The standard error of the Monte-Carlo approximation can be found by using the expression:  $\sqrt{\bar{p}(1 - \bar{p})/MC}$ , where  $\bar{p}$  is the coverage probability recorded.

## 4.2 Results

What follows are the results of the simulation study. The program was run according to the parameter values described in the previous section, and followed all the steps of the algorithm. All options concerning sample size, choice of the parameter  $m$ , method of calculating the percentile upper bound, distribution of the sample data, and the choice between calculating the mean or variance were considered and appear in the output.

### Example

The following table is an example of the output, where the confidence upper bound was calculated for the mean, the sample size was taken to be 20, the distribution of the sample data was taken to be the Standard Exponential distribution, and the method of calculating the percentile confidence upper bound was chosen to be the Backwards method ( $\alpha$  was selected to be 0.1):

Standard Exponential $n = 20$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	20	12.7898
$SE(\hat{m})$	0	0	0	0.050178
$E(U)$	1.281026	1.276291	1.278419	1.283863
$SE(U)$	0.00294745	0.00295453	0.0029325	0.00297965
Coverage Probability	0.8261	0.829	0.8347	0.8305
$SE(\text{Coverage Probability})$	0.00379	0.003765	0.003715	0.003752

The labels in the first column have the following meaning:

1. Choice of  $m$ : This refers to the manner in which the parameter  $m$  was chosen. The parameter  $m$  is used in the calculation of the modified bootstrap estimation of the percentile confidence upper bound. The parameter can be determined in one of two ways: The first is to simply make it a fraction of the sample size  $n$ . The second is to calculate it data-dependently using the methods found in Section 3.2. In the output, the first three columns refer to a choice of  $m$  as a fraction of the sample size. The fractions are one third the sample size, two thirds the sample size, and equal to the sample size. The last column refers to the  $m$  calculated data-dependently and is called  $\hat{m}$ .
2.  $E(\hat{m})$ : This refers to the expected value of  $\hat{m}$ . For the first three choices of  $m$ , this row reflects the fact that the choice of  $m$  is a fixed fraction of the sample size. The last column, however, shows the average of all the  $\hat{m}$ 's calculated under the simulation.
3.  $SE(\hat{m})$ : This row shows the standard error of the  $m$ 's calculated in the study. It is clear that the first three columns have a standard deviation of zero since they are non-random quantities. The last column gives an idea of the variation of the  $\hat{m}$ 's calculated in the study.
4.  $E(U)$ : The expected value of the 90% percentile confidence upper bound. Denote the upper bound by  $U$ , while the term  $E(U)$  simply refers to the average value of the upper bounds calculated in the study.
5.  $SE(U)$ : This refers to the standard error of the values of  $U$ . It gives an idea as to how much one can expect the value  $E(U)$  to vary from one simulation to the next.
6. Coverage Probability: The coverage probability is the estimate of how well the upper bound covers the true parameter. One would expect this value to be as close as possible to the specified coverage of  $1 - \alpha = 0.9$ .
7.  $SE(\text{Coverage Probability})$ : An estimate of the degree to which this coverage probability will vary from one simulation study to the next.

A full account of all the results can be found in Appendix A.

## 4.3 Conclusions

For the following conclusions attention will be restricted to the output derived from the Backwards percentile method. The reason for this is that it is believed that the Backwards method performs better than the Hybrid method in most cases (compare the results in Appendix A, Sections A.1 and A.3, the Backwards upper bounds, and Sections A.2 and A.4, the Hybrid upper bounds).

### 4.3.1 Confidence upper bound for the mean

The output of the simulation reveals a fact which is supported by the findings of Chung and Lee (2001). Chung and Lee, using a different criterion, or rule, for selecting the parameter  $m$ , came to the conclusion that when working with modified bootstrap percentile confidence bounds (or intervals) for the mean, the asymptotically (as  $n \rightarrow \infty$ ) optimal choice of  $m$  is independent of the sample data (a proof of this statement can be found in Chung and Lee(2001:231)). The results obtained from the simulations run in this study, making use of the Bickel, Götze and van Zwet method (1997), also suggests the same conclusion.

This means that when calculating the percentile confidence upper bound with the modified bootstrap, the size of the modified bootstrap sample can be very small ( as small as one quarter of the sample size  $n$ ) without affecting the results. This conclusion can be seen from the output. Consider any of the output tables in Appendix A Section A.1. The coverage probability for all choices of the tuning parameter  $m$  yields the same or similar results, and the difference between them can be attributed to simple stochastic variation.

The decision to make use of a data-dependent parameter  $\hat{m}$  for the percentile confidence upper bound mean problem, is moot. Since the choice of  $m$  is completely independent of the data one can simply choose  $m$  to be very small and still expect the same results. This data independence of the parameter  $m$  is reflected in the fact that the estimate  $\hat{m}$  usually took on a value between one half and two thirds of the sample size ( as the sample size increased the value drew closer to a half of the sample size).

It is clear from the output that not all of the simulations reach their target of a 90% coverage probability. However, it can also be seen that these values will eventually converge to the desired coverage as the sample size increases.

In conclusion, it is fair to say that a data-driven choice of  $m$  (in this particular problem) is unnecessary, because nearly any choice of  $m$  will be just as good. The result is that when calculating these modified bootstrap percentile confidence upper bounds or intervals, the computer processing time can be greatly reduced by choosing  $m$  to be any small fraction of the sample size  $n$ .

### 4.3.2 Confidence upper bound for the variance

Looking at the output one can once again see that the choice of  $m$  has little or no effect on the coverage probability. This agrees with the results found in the case of the mean and is not surprising if one considers the work done by Chung and Lee(2001). As mentioned above, they proved that, in the case of the mean, the asymptotically (as  $n \rightarrow \infty$ ) optimal choice of  $m$  is not data-dependent. However, the sample variance,  $s_n^2$ , can also be written (for large  $n$ ) as a sample mean, viz.

$$\begin{aligned}
 s_n^2 &= \frac{1}{n} \sum_{i=1}^n (X_i - \bar{X}_n)^2 \\
 &= \frac{1}{n} \sum_{i=1}^n (X_i - \mu)^2 - (\bar{X}_n - \mu)^2 \\
 &= \frac{1}{n} \sum_{i=1}^n (X_i - \mu)^2 + O_P(n^{-1}) \\
 &= \text{The mean of } (X_1 - \mu)^2, (X_2 - \mu)^2, \dots, (X_n - \mu)^2 \\
 &\quad \text{plus a negligible remaining term.}
 \end{aligned}$$

Hence, according to the results obtained by Chung and Lee (2001), the optimal choice of  $m$  (in the case of the variance) should also not depend on the data. This is in agreement with what was found by applying the, entirely different, data-based choice of Bickel, Götze and van Zwet (1997) (see Appendix A, Section A.3 for the variance results).

Unfortunately, the coverage probabilities that one sees in the output are a long way from the desired 90% (achieving instead anywhere in the region of 60% and 85%). These results do, however, tend to improve with an increase in the choice of the sample size,  $n$ . This outcome can be attributed to the fact that the variance is notoriously difficult to estimate accurately with small sample sizes.

The conclusion reached for the variance case is then the same as the conclusion for the mean: It is not necessary to make use of a data-dependent choice of  $m$  when calculating percentile confidence upper bounds for the variance, because almost all choices of  $m$ , whether selected arbitrarily or data-dependently, will yield very similar results. This of course means that if one were to calculate a percentile confidence upper bound for the variance using the modified bootstrap, the amount of computing time can be drastically reduced by simply choosing  $m$  to be some small fraction of the sample size  $n$ .

# Appendix A

A.1 Results for the confidence upper bounds for the mean using the Backwards method with  $\alpha = 0.1$

Standard Exponential $n = 20$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	20	12.79
$SD(\hat{m})$	0	0	0	0.05
$E(U)$	1.281	1.276	1.278	1.283
$SD(U)$	0.003	0.003	0.003	0.003
Coverage Probability	0.826	0.829	0.834	0.831
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

F(5,8) distribution $n = 20$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	20	13.06
$SD(\hat{m})$	0	0	0	0.05
$E(U)$	1.693	1.690	1.687	1.685
$SD(U)$	0.005	0.005	0.005	0.005
Coverage Probability	0.784	0.781	0.786	0.787
$SD(\text{Coverage Probability})$	0.004	0.004	0.004104055	0.004

Weibull(0.5, 1) distribution $n = 20$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	20	13.69
$SD(\hat{m})$	0	0	0	0.05
$E(U)$	3.078	3.146	3.021	3.024
$SD(U)$	0.017	0.019	0.016	0.016
Coverage Probability	0.724	0.723	0.708	0.711
$SD(\text{Coverage Probability})$	0.004	0.004	0.005	0.004534839

$0.75 N(0, 1) + 0.25 N(1, 0.01)$ $n = 20$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	20	12.54
$SD(\hat{m})$	0	0	0	0.05
$E(U)$	0.512	0.516	0.514	0.517
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.896	0.904	0.902	0.901
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

$0.5 N(0, 1) + 0.5 N(1, 0.01)$ $n = 20$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	20	13.87
$SD(\hat{m})$	0	0	0	0.05
$E(U)$	0.728	0.732	0.735	0.732
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.910	0.916	0.914	0.914
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

Standard Exponential $n = 50$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	27.66
$SD(\hat{m})$	0	0	0	0.14
$E(U)$	1.180	1.179	1.181	1.178
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.861	0.857	0.860	0.861
$SD(\text{Coverage Probability})$	0.003	0.004	0.003	0.003
F(5,8) distribution $n = 50$ Backwards Method				
	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	28.61
$SD(\hat{m})$	0	0	0	0.14
$E(U)$	1.576	1.578	1.574	1.573
$SD(U)$	0.003	0.003	0.003	0.003
Coverage Probability	0.820	0.828	0.829	0.828
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
Weibull(0.5, 1) distribution $n = 50$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	29.97
$SD(\hat{m})$	0	0	0	0.13
$E(U)$	2.757	2.765	2.757	2.741
$SD(U)$	0.010	0.010	0.010	0.009
Coverage Probability	0.784	0.783	0.781	0.789
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

0.75 N(0, 1) + 0.25 N(1, 0.01) $n = 50$ Backwards Method				
	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	26.76
$SD(\hat{m})$	0	0	0	0.14
$E(U)$	0.419	0.420	0.423	0.421
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.897	0.899	0.902	0.902
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 50$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	28.88
$SD(\hat{m})$	0	0	0	0.13
$E(U)$	0.653	0.650	0.653	0.651
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.913	0.910	0.910	0.908
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
Standard Exponential $n = 100$ Backwards Method				
	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	52.38
$SD(\hat{m})$	0	0	0	0.28
$E(U)$	1.131	1.128	1.128	1.129
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.881	0.874	0.874	0.871
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

F(5,8) distribution $n = 100$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	54.32
$SD(\hat{m})$	0	0	0	0.28
$E(U)$	1.509	1.508	1.507	1.507
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.849	0.845	0.842	0.848
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
Weibull(0.5, 1) distribution $n = 100$ Backwards Method				
	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	56.60
$SD(\hat{m})$	0	0	0	0.27
$E(U)$	2.555	2.567	2.550	2.556
$SD(U)$	0.006	0.006	0.006	0.006
Coverage Probability	0.821	0.822	0.812	0.819
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
$0.75 N(0, 1) + 0.25 N(1, 0.01)$ $n = 100$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	50.65
$SD(\hat{m})$	0	0	0	0.28
$E(U)$	0.372	0.372	0.372	0.373
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.901	0.904	0.900	0.903
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 100$ Backwards Method				
	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	52.66
$SD(\hat{m})$	0	0	0	0.27
$E(U)$	0.608	0.608	0.609	0.607
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.907	0.907	0.907	0.903
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
Standard Exponential $n = 200$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	103.21
$SD(\hat{m})$	0	0	0	0.57
$E(U)$	1.091	1.090	1.091	1.090
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.882	0.883	0.881	0.885
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
F(5,8) distribution $n = 200$ Backwards Method				
	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	104.76
$SD(\hat{m})$	0	0	0	0.56
$E(U)$	1.460	1.456	1.457	1.457
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.861	0.853	0.861	0.857
$SD(\text{Coverage Probability})$	0.003	0.004	0.003	0.004

Weibull(0.5, 1) distribution $n = 200$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	108.94
$SD(\hat{m})$	0	0	0	0.55
$E(U)$	2.401	2.400	2.396	2.401
$SD(U)$	0.004	0.004	0.004	0.004
Coverage Probability	0.842	0.842	0.840	0.841
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
0.75 N(0, 1) + 0.25 N(1, 0.01) $n = 200$ Backwards Method				
	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	100.12
$SD(\hat{m})$	0	0	0	0.58
$E(U)$	0.336	0.337	0.338	0.336
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.896	0.903	0.905	0.903
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 200$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	101.22
$SD(\hat{m})$	0	0	0	0.56
$E(U)$	0.577	0.578	0.577	0.577
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.908	0.902	0.905	0.904
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

## A.2 Results for the confidence upper bounds for the mean using the Hybrid method with $\alpha = 0.1$

Standard Exponential $n = 20$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	6	13.04
$SD(\hat{m})$	0	0	0	0.05
$E(U)$	1.249	1.258	1.249	1.254
$SD(U)$	0.003	0.003	0.003	0.003
Coverage Probability	0.809	0.810	0.809	0.810
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
F(5,8) distribution $n = 20$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	6	12.89
$SD(\hat{m})$	0	0	0	0.06
$E(U)$	1.641	1.656	1.641	1.636
$SD(U)$	0.004	0.004	0.004	0.004
Coverage Probability	0.765	0.773	0.765	0.765
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

Weibull(0.5, 1) distribution $n = 20$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	6	14.01
$SD(\hat{m})$	0	0	0	0.06
$E(U)$	2.834	2.949	2.834	2.832
$SD(U)$	0.015	0.016	0.015	0.014
Coverage Probability	0.682	0.694	0.682	0.684
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005
0.75 N(0, 1) + 0.25 N(1, 0.01) $n = 20$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	6	12.14
$SD(\hat{m})$	0	0	0	0.05
$E(U)$	0.528	0.520	0.528	0.525
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.909	0.898	0.909	0.906
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 20$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	6	13	6	12.79
$SD(\hat{m})$	0	0	0	0.05
$E(U)$	0.747	0.742	0.747	0.746
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.919	0.920	0.919	0.925
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

Standard Exponential $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	16	28.99
$SD(\hat{m})$	0	0	0	0.13
$E(U)$	1.169	1.176	1.169	1.167
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.846	0.852	0.846	0.847
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
F(5,8) distribution $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	16	30.42
$SD(\hat{m})$	0	0	0	0.14
$E(U)$	1.544	1.560	1.544	1.546
$SD(U)$	0.003	0.003	0.003	0.003
Coverage Probability	0.802	0.815	0.802	0.810
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
Weibull(0.5, 1) distribution $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	16	34.81
$SD(\hat{m})$	0	0	0	0.13
$E(U)$	2.635	2.682	2.635	2.626
$SD(U)$	0.009	0.009	0.009	0.009
Coverage Probability	0.754	0.769	0.754	0.752
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

0.75 N(0, 1) + 0.25 N(1, 0.01) $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	16	26.77
$SD(\hat{m})$	0	0	0	0.14
$E(U)$	0.428	0.423	0.428	0.426
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.908	0.903	0.908	0.904
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.002947283

0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	16	27.38
$SD(\hat{m})$	0	0	0	0.14
$E(U)$	0.658	0.658	0.658	0.658
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.914	0.920	0.914	0.918
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
Standard Exponential $n = 100$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	33	53.78
$SD(\hat{m})$	0	0	0	0.27
$E(U)$	1.124	1.124	1.124	1.122
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.872	0.870	0.872	0.859
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

F(5,8) distribution $n = 100$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	33	58.23
$SD(\hat{m})$	0	0	0	0.27
$E(U)$	1.494	1.498	1.494	1.495
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.831	0.845	0.831	0.833
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
Weibull(0.5, 1) distribution $n = 100$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	33	67.02
$SD(\hat{m})$	0	0	0	0.26
$E(U)$	2.486	2.496	2.486	2.490
$SD(U)$	0.006	0.006	0.006	0.006
Coverage Probability	0.797	0.789	0.797	0.798
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
$0.75N(0, 1) + 0.25N(1, 0.01)$ $n = 100$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	33	51.20
$SD(\hat{m})$	0	0	0	0.29
$E(U)$	0.375	0.374	0.375	0.375
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.907	0.905	0.907	0.907
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 100$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	33	52.68
$SD(\hat{m})$	0	0	0	0.28
$E(U)$	0.614	0.612	0.614	0.611
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.915	0.916	0.915	0.915
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
Standard Exponential $n = 200$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	66	102.22
$SD(\hat{m})$	0	0	0	0.56
$E(U)$	1.088	1.088	1.089	1.088
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.881	0.874	0.881	0.874
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
F(5,8) distribution $n = 200$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	66	110.14
$SD(\hat{m})$	0	0	0	0.54
$E(U)$	1.451	1.453	1.451	1.447
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.847	0.852	0.847	0.843
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

Weibull(0.5, 1) distribution $n = 200$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	66	124.6822
$SD(\hat{m})$	0	0	0	0.49
$E(U)$	2.368	2.372	2.368	2.366
$SD(U)$	0.004	0.004	0.004	0.004
Coverage Probability	0.825	0.827	0.825	0.822
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
0.75 N(0, 1) + 0.25 N(1, 0.01) $n = 200$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	66	98.77
$SD(\hat{m})$	0	0	0	0.57
$E(U)$	0.339	0.338	0.339	0.338
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.904	0.902	0.904	0.906
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 200$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	66	101.37
$SD(\hat{m})$	0	0	0	0.57
$E(U)$	0.580	0.579	0.580	0.579
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.914	0.908	0.914	0.907
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

### A.3 Results for the confidence upper bounds for the variance using the Backwards method with $\alpha = 0.1$

Standard Exponential $n = 50$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	32.34
$SD(\hat{m})$	0	0	0	0.14
$E(U)$	1.421	1.414	1.396	1.387
$SD(U)$	0.007	0.007	0.006	0.006
Coverage Probability	0.723	0.712	0.709	0.716
$SD(\text{Coverage Probability})$	0.004	0.005	0.005	0.005
F(5,8) distribution $n = 50$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	30.75
$SD(\hat{m})$	0	0	0	0.15
$E(U)$	3.348	3.344	3.050	2.691
$SD(U)$	0.076	0.063	0.045	0.027
Coverage Probability	0.545	0.543	0.527	0.515
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

Weibull(0.5, 1) distribution $n = 50$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	30.29
$SD(\hat{m})$	0	0	0	0.17
$E(U)$	35.732	37.419	34.650	29.054
$SD(U)$	0.514	0.598	0.491	0.332
Coverage Probability	0.531	0.527	0.518	0.477
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

$0.75 N(0, 1) + 0.25 N(1, 0.01)$ $n = 50$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	31.36
$SD(\hat{m})$	0	0	0	0.136
$E(U)$	1.172	1.159	1.154	1.170
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.835	0.816	0.814	0.827
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

$0.5 N(0, 1) + 0.5 N(1, 0.01)$ $n = 50$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	31.52
$SD(\hat{m})$	0	0	0	0.13
$E(U)$	0.984	0.975	0.968	0.982
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.823	0.816	0.808	0.826
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

Standard Exponential $n = 100$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{n}$
$E(\hat{n})$	33	66	100	58.62
$SD(\hat{n})$	0	0	0	0.28
$E(U)$	1.336	1.323	1.314	1.314
$SD(U)$	0.004	0.005	0.004	0.004
Coverage Probability	0.779	0.766	0.763	0.777
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
F(5,8) distribution $n = 100$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{n}$
$E(\hat{n})$	33	66	100	55.76
$SD(\hat{n})$	0	0	0	0.31
$E(U)$	2.977	3.101	2.973	2.618
$SD(U)$	0.036	0.045	0.050	0.026
Coverage Probability	0.592	0.590	0.588	0.564
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

Weibull(0.5, 1) distribution $n = 100$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{n}$
$E(\hat{n})$	33	66	100	55.67
$SD(\hat{n})$	0	0	0	0.32
$E(U)$	33.868	34.381	33.066	28.537
$SD(U)$	0.364	0.385	0.336	0.237
Coverage Probability	0.588	0.595	0.589	0.553
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

0.75 N(0, 1) + 0.25 N(1, 0.01) $n = 100$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	55.29
$SD(\hat{m})$	0	0	0	0.28
$E(U)$	1.109	1.098	1.099	1.109
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.859	0.842	0.849	0.851
$SD(\text{Coverage Probability})$	0.003	0.004	0.004	0.004
0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 100$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	56.37
$SD(\hat{m})$	0	0	0	0.27
$E(U)$	0.918	0.914	0.913	0.921
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.849	0.844	0.844	0.852
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
Standard Exponential $n = 200$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	113.42
$SD(\hat{m})$	0	0	0	0.55
$E(U)$	1.240	1.236	1.240	1.237
$SD(U)$	0.003	0.003	0.003	0.003
Coverage Probability	0.812	0.800	0.804	0.804
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

F(5,8) distribution $n = 200$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	110.83
$SD(\hat{m})$	0	0	0	0.60
$E(U)$	2.846	2.935	2.858	2.574
$SD(U)$	0.032	0.039	0.058	0.015
Coverage Probability	0.642	0.638	0.637	0.620
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

Weibull(0.5, 1) distribution $n = 200$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	113.12
$SD(\hat{m})$	0	0	0	0.61
$E(U)$	31.405	32.568	30.832	28.247
$SD(U)$	0.258	0.294	0.234	0.179
Coverage Probability	0.652	0.655	0.642	0.624
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

$0.75 N(0, 1) + 0.25 N(1, 0.01)$ $n = 200$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	106.51
$SD(\hat{m})$	0	0	0	0.57
$E(U)$	1.058	1.053	1.057	1.060
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.870	0.861	0.868	0.874
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 200$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{n}$
$E(\hat{n})$	66	133	200	107.33
$SD(\hat{n})$	0	0	0	0.56
$E(U)$	0.871	0.871	0.869	0.873
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.859	0.863	0.862	0.867
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
Standard Exponential $n = 300$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{n}$
$E(\hat{n})$	100	200	300	165.85
$SD(\hat{n})$	0	0	0	0.837
$E(U)$	1.202	1.200	1.200	1.200
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.826	0.819	0.823	0.824
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
F(5,8) distribution $n = 300$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{n}$
$E(\hat{n})$	100	200	300	167.77
$SD(\hat{n})$	0	0	0	0.88
$E(U)$	2.746	2.852	2.664	2.543
$SD(U)$	0.037	0.063	0.018	0.015
Coverage Probability	0.655	0.652	0.653	0.642
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

Weibull(0.5, 1) distribution $n = 300$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	100	200	300	171.59
$SD(\hat{m})$	0	0	0	0.87
$E(U)$	29.761	30.310	29.770	28.319
$SD(U)$	0.205	0.217	0.210	0.169
Coverage Probability	0.681	0.677	0.674	0.671
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005
0.75 N(0, 1) + 0.25 N(1, 0.01) $n = 300$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	100	200	300	158.46
$SD(\hat{m})$	0	0	0	0.86
$E(U)$	1.036	1.035	1.034	1.039
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.872	0.871	0.869	0.873
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 300$ Backwards Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	100	200	300	157.53
$SD(\hat{m})$	0	0	0	0.86
$E(U)$	0.851	0.849	0.852	0.853
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.873	0.864	0.871	0.877
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

A.4 Results for the confidence upper bounds for the variance using the Hybrid method with  $\alpha = 0.1$

Standard Exponential $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	32.24
$SD(\hat{m})$	0	0	0	0.16
$E(U)$	1.349	1.412	1.415	1.357
$SD(U)$	0.006	0.006	0.007	0.006
Coverage Probability	0.700	0.727	0.725	0.696
$SD(\text{Coverage Probability})$	0.005	0.004	0.004	0.005
F(5,8) distribution $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	28.19
$SD(\hat{m})$	0	0	0	0.18
$E(U)$	2.770	3.053	3.174	2.683
$SD(U)$	0.032	0.064	0.099	0.039
Coverage Probability	0.509	0.522	0.538	0.502
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

Weibull(0.5, 1) distribution $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	26.00
$SD(\hat{m})$	0	0	0	0.19
$E(U)$	30.00	33.95	34.54	27.53
$SD(U)$	0.411	0.502	0.488	0.327
Coverage Probability	0.472	0.514	0.517	0.453
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005
0.75 N(0, 1) + 0.25 N(1, 0.01) $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	28.48
$SD(\hat{m})$	0	0	0	0.14
$E(U)$	1.170	1.176	1.184	1.166
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.831	0.843	0.845	0.833
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 50$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	16	33	50	30.77
$SD(\hat{m})$	0	0	0	0.14
$E(U)$	0.977	0.987	0.989	0.980
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.824	0.821	0.837	0.820
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

Standard Exponential $n = 100$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	63.20
$SD(\hat{m})$	0	0	0	0.29
$E(U)$	1.282	1.304	1.308	1.281
$SD(U)$	0.004	0.004	0.004	0.004
Coverage Probability	0.750	0.772	0.771	0.749
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
F(5,8) distribution $n = 100$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	57.93
$SD(\hat{m})$	0	0	0	0.36
$E(U)$	2.723	2.837	2.913	2.536
$SD(U)$	0.028	0.032	0.032	0.024
Coverage Probability	0.565	0.567	0.591	0.544
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

Weibull(0.5, 1) distribution $n = 100$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	53.50
$SD(\hat{m})$	0	0	0	0.39
$E(U)$	29.176	31.461	32.254	26.591
$SD(U)$	0.280	0.315	0.341	0.228
Coverage Probability	0.547	0.566	0.571	0.517
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

0.75 N(0, 1) + 0.25 N(1, 0.01)				
$n = 100$				
Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	52.18
$SD(\hat{m})$	0	0	0	0.29
$E(U)$	1.107	1.109	1.108	1.103
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.860	0.862	0.857	0.852
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.004
0.5 N(0, 1) + 0.5 N(1, 0.01)				
$n = 100$				
Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	33	66	100	54.18
$SD(\hat{m})$	0	0	0	0.28
$E(U)$	0.919	0.921	0.923	0.916
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.856	0.858	0.859	0.850
$SD(\text{Coverage Probability})$	0.004	0.003	0.003	0.004
Standard Exponential				
$n = 200$				
Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	125.63
$SD(\hat{m})$	0	0	0	0.54
$E(U)$	1.225	1.230	1.23	1.211
$SD(U)$	0.003	0.003	0.003	0.003
Coverage Probability	0.800	0.801	0.810	0.784
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004

F(5,8) distribution $n = 200$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	117.14
$SD(\hat{m})$	0	0	0	0.71
$E(U)$	2.618	2.721	2.775	2.412
$SD(U)$	0.025	0.052	0.027	0.017
Coverage Probability	0.605	0.616	0.621	0.577
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

Weibull(0.5, 1) distribution $n = 200$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	113.68
$SD(\hat{m})$	0	0	0	0.76
$E(U)$	28.353	29.689	29.910	25.982
$SD(U)$	0.193	0.235	0.225	0.164
Coverage Probability	0.621	0.625	0.633	0.578
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

$0.75 N(0, 1) + 0.25 N(1, 0.01)$ $n = 200$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	100.57
$SD(\hat{m})$	0	0	0	0.58
$E(U)$	1.058	1.059	1.061	1.059
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.868	0.870	0.875	0.867
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

0.5 N(0, 1) + 0.5 N(1, 0.01) $n = 200$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	66	133	200	103.07
$SD(\hat{m})$	0	0	0	0.57
$E(U)$	0.874	0.872	0.874	0.870
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.870	0.864	0.871	0.863
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003
Standard Exponential $n = 300$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	100	200	300	184.40
$SD(\hat{m})$	0	0	0	0.81
$E(U)$	1.183	1.193	1.195	1.182
$SD(U)$	0.002	0.002	0.002	0.002
Coverage Probability	0.807	0.824	0.824	0.806
$SD(\text{Coverage Probability})$	0.004	0.004	0.004	0.004
F(5,8) distribution $n = 300$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	100	200	300	176.14
$SD(\hat{m})$	0	0	0	1.05
$E(U)$	2.560	2.702	2.650	2.398
$SD(U)$	0.019	0.058	0.020	0.017
Coverage Probability	0.622	0.647	0.638	0.602
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

Weibull(0.5, 1) distribution $n = 300$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	100	200	300	176.94
$SD(\hat{m})$	0	0	0	1.11
$E(U)$	27.429	28.557	28.946	25.642
$SD(U)$	0.165	0.180	0.203	0.128
Coverage Probability	0.638	0.661	0.662	0.617
$SD(\text{Coverage Probability})$	0.005	0.005	0.005	0.005

$0.75 N(0, 1) + 0.25 N(1, 0.01)$ $n = 300$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	100	200	300	149.10
$SD(\hat{m})$	0	0	0	0.88
$E(U)$	1.035	1.039	1.037	1.035
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.875	0.881	0.873	0.877
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003279775

$0.5 N(0, 1) + 0.5 N(1, 0.01)$ $n = 300$ Hybrid Method				
Choice of $m$	$n/3$	$2n/3$	$n$	$\hat{m}$
$E(\hat{m})$	100	200	300	152.01
$SD(\hat{m})$	0	0	0	0.86
$E(U)$	0.850	0.852	0.853	0.850
$SD(U)$	0.001	0.001	0.001	0.001
Coverage Probability	0.872	0.875	0.874	0.871
$SD(\text{Coverage Probability})$	0.003	0.003	0.003	0.003

## Appendix B

# Fortran implementation of the data dependent choice of $m$

The FORTRAN implementation of the algorithm described in Chapter 4 can be found on the pages that follow. The program receives input regarding the choice of the parameter for which the confidence upper bound is to be constructed (either the mean or the variance), the setting of the  $\alpha$  value (which was set to 0.1 in all cases), and the choice of which distribution to generate the data from. The program outputs four separate text files, namely PROJOUT1.TXT, PROJOUT2.TXT, PROJOUT3.TXT and PROJOUT4MHAT.TXT. These four outputs correspond to the way in which the parameter  $m$  is estimated, i.e. PROJOUT1.TXT corresponds to the choice of  $m = n/3$ , PROJOUT2.TXT corresponds to  $m = 2n/3$ , PROJOUT3.TXT corresponds to  $m = n$  and PROJOUT4MHAT.TXT corresponds to the data-dependent choice,  $\hat{m}$ .

In addition to the main program there is also included here a file populated with the functions and subroutines used in the program.

```
include 'include.f90'
```

```
program Project
```

```
use MODFUNCSANDSUBS
```

```
implicit none
```

```
INTEGER*4 B_reps,MC_reps, b, i, M_TOT, msize, mc, nsize, TMPINT,j,third_n
```

```
REAL*8 INCRM, ALPHA
```

```
PARAMETER( B_reps = 1000, MC_reps = 10000, INCRM = 2.0d0)
```

```
INTEGER*4,ALLOCATABLE::M_GRID(:)
```

```
INTEGER*4,ALLOCATABLE::INDEX_GRID(:)
```

```
INTEGER*4 M_HAT
```

```
REAL*8 tmp1, tmp2, tmp_s
```

```
REAL*8, ALLOCATABLE::A_TEST_STAT1(:)
```

```
REAL*8, ALLOCATABLE::A_TEST_STAT2(:)
```

```
REAL*8 DegFree1, Degfree2
```

```
REAL*8 ACTUALPARAM, ESTPARAM
```

```
REAL*8, ALLOCATABLE:: DATAVEC(:)
```

```
REAL*8, ALLOCATABLE::DATAMATRIXSTAR1(:,:)
```

```
REAL*8, ALLOCATABLE::DATAMATRIXSTAR2(:,:)
```

```
REAL*8 BQGTSTATVEC1(B_reps), BOOTSTATVEC2(B_reps),BOOTSTATVEC(B_reps)
```

```
REAL*8 UPPER_BOUND, UPPER_BOUND_TEST
```

```
INTEGER*4, ALLOCATABLE::INDEX_SWR(:)
```

```
REAL*8, ALLOCATABLE::TEMPVEC1(:)
```

```
REAL*8, ALLOCATABLE::TEMPVEC2(:)
```

```
INTEGER*4 N(4), counter, CONTAM_COUNT, PROGRESS_COUNTER
```

```
REAL*8 UPPER_VECTOR(MC_Reps),UPPER_VECTOR_TEST(MC_Reps), M_HAT_VECTOR(MC_Reps)
```

```
INTEGER*4 CHOICE_CONFBNB, CHOICE_HOW2GET_M, CHOICE_DISTRIB, CHOICE_PARAM
```

```
REAL*8 MU1, MU2, SIGMA1, SIGMA2, CONTAM_PROB(3)
```

```
REAL*8 PROPORTION, PROPORTION_TEST
```

```
PRINT*, " Which parameter do you want to use?:"
```

```
PRINT*, " 1) Mean"
```

```
PRINT*, " 2) Variance"
```

```

READ(*,*) CHOICE_PARAM
!CHOICE_PARAM = 1
PRINT*, " What should Alpha be?:"
READ(*,*) ALPHA
PRINT*, " What Distribution?:"
PRINT*, " 1) Standard Exponential"
PRINT*, " 2) F (5,8)"
PRINT*, " 3) Weibul (0.5)"
PRINT*, " 4) Standard Normal N(0,1)"
PRINT*, " 5) Contaminated Normal (1-0.25)*N(0,1) + (0.25)*N(1,0.01)"
PRINT*, " 6) Contaminated Normal (1-0.5)*N(0,1) + (0.5)*N(1,0.01)"
READ(*,*) CHOICE_DISTRIB

!For displaying progress
PROGRESS_COUNTER = 0

!Setup Contamination probabilities
CONTAM_PROB = (/ 0.0d0, 0.25d0, 0.5d0 /)

!Setup N: Sample sizes
IF (CHOICE_PARAM .EQ. 1) THEN
N = (/ 20, 50, 100, 200 /)
ELSE
N = (/ 50, 100, 200, 300 /)
ENDIF

!Initialize parameters
MU1 = 0d0
MU2 = 1.0d0
SIGMA1 = 1.0d0
SIGMA2 = 0.1d0
Degfree1 = 5.0d0 !for F distribution
Degfree2 = 8.0d0 !for F distribution
DO third_n = 1,4
!Start writing to the file
IF(third_n .EQ. 1) THEN
OPEN(1, FILE = "PROJOUT1.TXT")

```

```

ELSEIF(third_n .EQ. 2) THEN
OPEN(1, FILE = "PROJOUT2.TXT")
ELSEIF(third_n .EQ. 3) THEN
OPEN(1, FILE = "PROJOUT3.TXT")
ELSEIF(third_n .EQ. 4) THEN
OPEN(1, FILE = "PROJOUT4MHAT.TXT")
ENDIF

WRITE(1,*) "*****"
IF (CHOICE_PARAM .EQ. 1) THEN
WRITE(1,*) "Calculating_the_sample_Mean"
ELSEIF (CHOICE_PARAM .EQ. 2) THEN
WRITE(1,*) "Calculating_the_sample_Variance"
ENDIF

WRITE(1,*) "Alpha=", ALPHA
WRITE(1,*) "MC_Reps=", MC_Reps
WRITE(1,*) "B_Reps=", B_Reps
WRITE(1,*) "MU1=", MU1
WRITE(1,*) "MU2=", MU2
WRITE(1,*) "SIGMA1=", SIGMA1
WRITE(1,*) "SIGMA2=", SIGMA2
WRITE(1,*) "Format:"
WRITE(1,*) "-----"
WRITE(1,*) "Distribution of Data"
WRITE(1,*) "N:Sample_size"
WRITE(1,*) "Conf.int.type:Hybrid/Back"
WRITE(1,*) "E(m_hat)"
WRITE(1,*) "SD(m_hat)"
WRITE(1,*) "E(Upper)"           !Upperbound using modified bootstrap with m_hat
WRITE(1,*) "SD(Upper)"           !Upperbound using modified bootstrap with m_hat
WRITE(1,*) "PROPORTION"         !Proportion using modified bootstrap with m_hat
WRITE(1,*) "SD(PROPORTION)"     !Proportion using modified bootstrap with m_hat
WRITE(1,*) "*****"

!Begin main outer loop for the sample sizes
DO nsize=1,4

!Allocate data vector with specified sample size

```

```

ALLOCATE(DATAVEC(N(nsize)))

!Allocate a matrix of size (B_reps * N(nsize)) for the bootstrap replications
ALLOCATE(DATAMATRIXSTAR1(B_reps, N(nsize)))
ALLOCATE(DATAMATRIXSTAR2(B_reps, N(nsize)))
ALLOCATE(INDEX_SWR(N(nsize)))

!Begin inner loops over the different methods of confidence intervals,
!ways of getting m, and ways in which to generate data
DO CHOICE_HOW2GET_M = 1, 2
!Initialise the proportion
PROPORTION = 0d0
PROPORTION_TEST = 0d0

DO mc = 1, MC_reps !Monte Carlo Loop
!Generate new data for each Monte Carlo
IF (CHOICE_PARAM .EQ. 1) THEN
IF( CHOICE_DISTRIB .EQ. 1 ) THEN
CALL DRNEXP(N(nsize), DATAVEC)
ACTUALPARAM = 1
ELSEIF ( CHOICE_DISTRIB .EQ. 2 ) THEN
CALL DF_VARIATES(N(nsize),Degfree1,Degfree2,DATAVEC)
ACTUALPARAM = Degfree2/(Degfree2-2)
ELSEIF ( CHOICE_DISTRIB .EQ. 3 ) THEN
CALL DRNWIB(N(nsize),0.5d0,DATAVEC)
!Weibull shape parameter = 0.5, scale parameter = 1
ACTUALPARAM = 2.0d0
ELSEIF ( CHOICE_DISTRIB .EQ. 4 ) THEN
CALL DNORMALVARIATES( N(nsize),MU1,SIGMA1,DATAVEC)
ACTUALPARAM = MU1
ELSEIF ( CHOICE_DISTRIB .EQ. 5 ) THEN
CALL DCONTAMNORMALVARIATES ( N(nsize), MU1, SIGMA1, MU2, SIGMA2, 0.25d0, DATAVEC)
ACTUALPARAM = (1-0.25d0)*MU1 + (0.25d0)*MU2
ELSEIF ( CHOICE_DISTRIB .EQ. 6 ) THEN
CALL DCONTAMNORMALVARIATES ( N(nsize), MU1, SIGMA1, MU2, SIGMA2, 0.5d0, DATAVEC)
ACTUALPARAM = (1-0.5d0)*MU1 + (0.5d0)*MU2
ENDIF
ESTPARAM = DCALC_MEAN(N(nsize), DATAVEC)

```

```

ELSEIF (CHOICE_PARAM .EQ. 2) THEN
IF( CHOICE_DISTRIB .EQ. 1 ) THEN
!Standard Exponential (lambda = 1)
CALL DRNEXP(N(nsize), DATAVEC)
ACTUALPARAM = 1
ELSEIF ( CHOICE_DISTRIB .EQ. 2 ) THEN
!F distribution
CALL DF_VARIATES(N(nsize),Degfree1,Degfree2,DATAVEC)
ACTUALPARAM =(2*(Degfree2**2)*(Degfree1+Degfree2-2))
& /(Degfree1*(Degfree2-4)*((Degfree2-2)**2))
ELSEIF ( CHOICE_DISTRIB .EQ. 3 ) THEN
!Weibull with shape parameter = 0.5, scale parameter = 1
CALL DRNWIB(N(nsize),0.5d0,DATAVEC)
ACTUALPARAM = 20.0d0
ELSEIF ( CHOICE_DISTRIB .EQ. 4 ) THEN
CALL DNORMALVARIATES( N(nsize),MU1,SIGMA1,DATAVEC)
ACTUALPARAM = SIGMA1**2
ELSEIF ( CHOICE_DISTRIB .EQ. 5 ) THEN
CALL DCONTAMNORMALVARIATES ( N(nsize), MU1, SIGMA1, MU2, SIGMA2, 0.25d0, DATAVEC)
ACTUALPARAM=(SIGMA1**2)*(0.75d0)+(SIGMA2**2)*(0.25d0)+(0.75d0*0.25d0)*((MU1-MU2)**2)
ELSEIF ( CHOICE_DISTRIB .EQ. 6 ) THEN
CALL DCONTAMNORMALVARIATES ( N(nsize), MU1, SIGMA1, MU2, SIGMA2, 0.5d0, DATAVEC)
ACTUALPARAM = (SIGMA1**2)*(0.5d0) + (SIGMA2**2)*(0.5d0) + (0.25d0)*((MU1 - MU2)**2)
ENDIF
ESTPARAM = DCALC_SAMPLEVAR(N(nsize), DATAVEC)
ENDIF

!Generate your Bootstrap samples from the data generated:
DO j = 1, B_reps
CALL RNUND(N(nsize),N(nsize),INDEX_SWR)
DATAMATRIXSTAR1(j,:)= DATAVEC(INDEX_SWR)
ENDDO

IF (third_n .EQ. 4) THEN
!Set the number of "m"s to be half the size of the sample
!Then the grid will increment in 2's.
!This is because of the "divide by 2" (m/2) section of the

```

```

!"selection of m_hat". Starting at 2 eliminates the
!occurrence of sample sizes of zero.
M_TOT = INT(N(nsize)/2)
ALLOCATE(M_GRID(M_TOT))
ALLOCATE(INDEX_GRID(M_TOT))
ALLOCATE(A_TEST_STAT1(M_TOT))
ALLOCATE(A_TEST_STAT2(M_TOT))
!Setup the grid of m values (m being the modified sample size)
DO i = 1, M_TOT
M_GRID(i) = i*INCRM
ENDDO
!Loop over the m values to find one which minimises Am
!Using the matrix method
DO msize = 1, M_TOT !m-size loop
!Used in getting the minimum...see sub DSVRGP
DO counter = 1, M_TOT
INDEX_GRID(counter) = counter
ENDDO
!Then we use a different loop from choices 1 and 2
!Do modified bootstrap with "m=m"
!i.e. in Bootstatvec1 get theta_m*(1), theta_m*(2) ... theta_m*(B)
TMPINT = M_GRID(msize)
BOOTSTATVEC1 = 0
DO b = 1, B_reps !Bootloop 1
IF (CHOICE_PARAM .EQ. 1) THEN
BOOTSTATVEC1(b) = DCALC_MEAN(M_GRID(msize), DATAMATRIXSTAR1(b,1:M_GRID(msize)))
ELSEIF (CHOICE_PARAM .EQ. 2) THEN
BOOTSTATVEC1(b) = DCALC_SAMPLEVAR(M_GRID(msize), DATAMATRIXSTAR1(b,1:M_GRID(msize)))
ENDIF
ENDDO !end Bootloop 1
!Do modified bootstrap with "m=m/2"
!i.e. Bootstatvec2 get theta_m/2*(1), theta_m/2*(2) ... theta_m/2*(B)
TMPINT = INT( M_GRID(msize)/2 )
BOOTSTATVEC2 = 0
DO b = 1, B_reps !Bootloop 2
IF (CHOICE_PARAM .EQ. 1) THEN
BOOTSTATVEC2(b) = DCALC_MEAN(TMPINT, DATAMATRIXSTAR1(b,1:TMPINT))

```

```

ELSEIF (CHOICE_PARAM .EQ. 2) THEN
BOOTSTATVEC2(b) = DCALC_SAMPLEVAR(TMPINT, DATA MATRIX STAR1(b,1:TMPINT))
ENDIF
ENDDO !end Bootloop 2

!Sort the vectors
CALL DSVRGN(B_Reps,BOOTSTATVEC1,BOOTSTATVEC1)
CALL DSVRGN(B_Reps,BOOTSTATVEC2,BOOTSTATVEC2)
IF (CHOICE_HOW2GET_M .EQ. 1) THEN
!get the A test stat with method 1
A_TEST_STAT1(msize)=DCALC_A_STAT1(ALPHA,ESTPARAM,B_reps,BOOTSTATVEC1,B_reps,
& BOOTSTATVEC2, M_GRID(msize))
ELSEIF(CHOICE_HOW2GET_M .EQ. 2) THEN
!get the A test stat with method 2
A_TEST_STAT2(msize)=DCALC_A_STAT2(ALPHA,ESTPARAM,B_reps,BOOTSTATVEC1,B_reps,
& BOOTSTATVEC2, M_GRID(msize))
ENDIF
ENDDO !end m-size loop

IF (CHOICE_HOW2GET_M .EQ. 1) THEN
CALL DSVRGP(M_TOT, A_TEST_STAT1, A_TEST_STAT1, INDEX_GRID)
ELSEIF(CHOICE_HOW2GET_M .EQ. 2) THEN
CALL DSVRGP(M_TOT, A_TEST_STAT2, A_TEST_STAT2, INDEX_GRID)
ENDIF

!Get the value of m which minimised the A statistic under method 1
M_HAT = M_GRID(INDEX_GRID(1))
M_HAT_VECTOR(mc) = M_HAT
DEALLOCATE(M_GRID)
DEALLOCATE(INDEX_GRID)
DEALLOCATE(A_TEST_STAT1)
DEALLOCATE(A_TEST_STAT2)
ELSE
M_HAT = INT((third_n)*N(msize)/3)
M_HAT_VECTOR(mc) = M_HAT
ENDIF

!We now have m_hat!
!Calculate the confidence bound using

```

```

!m_hat via the modified bootstrap:
!=====
ALLOCATE(TEMPVEC1(M_HAT))
TMPINT = M_HAT
BOOTSTATVEC1 = 0
DO b=1, B_Reps
IF (CHOICE_PARAM .EQ. 1) THEN
TEMPVEC1 = DATAMATRIXSTAR1(b , 1:M_HAT)
BOOTSTATVEC1(b) = DCALC_MEAN( M_HAT , TEMPVEC1 )
ELSEIF (CHOICE_PARAM .EQ. 2) THEN
TEMPVEC1 = DATAMATRIXSTAR1(b , 1:M_HAT)
BOOTSTATVEC1(b) = DCALC_SAMPLEVAR( M_HAT , TEMPVEC1 )
ENDIF
ENDDO

CALL DSVRGN(B_Reps,BOOTSTATVEC1,BOOTSTATVEC1)
IF (CHOICE_HOW2GET_M .EQ. 1) THEN
!HYBRID METHOD
UPPER_BOUND = ESTPARAM - ((BOOTSTATVEC1(INT(B_Reps*ALPHA))-
& ESTPARAM)*(DSQRT(DFLOAT(M_HAT)) / DSQRT(DFLOAT(N(nsize))))))
ELSEIF (CHOICE_HOW2GET_M .EQ. 2) THEN
!BACKWARDS METHOD
UPPER_BOUND = ESTPARAM + (BOOTSTATVEC1(INT( B_Reps*(1-ALPHA)))
& ESTPARAM)*(DSQRT(DFLOAT(M_HAT))/DSQRT(DFLOAT(N(nsize))))
ENDIF
UPPER_VECTOR(mc) = UPPER_BOUND
DEALLOCATE(TEMPVEC1)

IF (ACTUALPARAM .LE. UPPER_BOUND) THEN
PROPORTION = PROPORTION + 1
ENDIF
PROGRESS_COUNTER = PROGRESS_COUNTER + 1
PRINT*, "Percentage complete: ",(PROGRESS_COUNTER/(FLOAT(MC_REPS)*4*2*4))*100, "%"
ENDDO !end Monte Carlo Loop
PROPORTION = PROPORTION/MC_Reps
IF (third_n .EQ. 1) THEN
WRITE(1,*) "n/3"

```

```

ELSEIF (third_n .EQ. 2) THEN
WRITE(1,*) "2n/3"
ELSEIF (third_n .EQ. 3) THEN
WRITE(1,*) "n"
ELSEIF (third_n .EQ. 4) THEN
WRITE(1,*) "m-hat"
ENDIF
IF (CHOICE_DISTRIB .EQ. 1) THEN
WRITE(1,*) "StdExp"
ELSEIF(CHOICE_DISTRIB .EQ. 2) THEN
WRITE(1,*) "F(5, 8)"
ELSEIF(CHOICE_DISTRIB .EQ. 3) THEN
WRITE(1,*) "Weibull(0.5)"
ELSEIF(CHOICE_DISTRIB .EQ. 4) THEN
WRITE(1,*) "N(0, 1)"
ELSEIF(CHOICE_DISTRIB .EQ. 5) THEN
WRITE(1,*) "0.75N(0, 1)+0.25N(1,0.01)"
ELSEIF(CHOICE_DISTRIB .EQ. 6) THEN
WRITE(1,*) " 0.5N(0, 1)+ 0.5N(1,0.01)"
ENDIF
WRITE(1,*) N(nsize)
IF (CHOICE_HOW2GET_M .EQ. 1) THEN
WRITE(1,*) "Hybrid"
ELSEIF (CHOICE_HOW2GET_M .EQ. 2) THEN
WRITE(1,*) "Back"
ENDIF
WRITE(1,*) "-----"
WRITE(1,*) DCALC_MEAN(MC_Reps,M_HAT_VECTOR)
WRITE(1,*) DCALC_STNDERROR(MC_Reps,M_HAT_VECTOR)/DSQRT(DFLOAT(MC_Reps))
WRITE(1,*) DCALC_MEAN(MC_Reps,UPPER_VECTOR)
WRITE(1,*) DCALC_STNDERROR(MC_Reps,UPPER_VECTOR)/DSQRT(DFLOAT(MC_Reps))
WRITE(1,*) PROPORTION
WRITE(1,*) DSQRT(PROPORTION*(1-PROPORTION)/DFLOAT(MC_Reps))
WRITE(1,*) "*****"
WRITE(1,*) "*****"

ENDDO !End Method 1 or 2 loop

```

```

MODULE MODFUNCSANDSUBS
CONTAINS
REAL*8 FUNCTION DCALC_A_STAT1(ALPHA,ESTPARAM, N1 ,VECTOR1, N2, VECTOR2, M)
!Double Version
!Get the A statistic using method 1
INTEGER*4 N1, N2, M
REAL*8 ALPHA, ESTPARAM, VECTOR1(N1), VECTOR2(N2), tmp1, tmp2
tmp1 = VECTOR1(FLOOR((ALPHA)*N1))
tmp2 = VECTOR2(FLOOR((ALPHA)*N2))
DCALC_A_STAT1=ABS((DSQRT(DFLOAT(M))*(tmp1-ESTPARAM))-(DSQRT(DFLOAT(M)/2)*
& (tmp2-ESTPARAM)))
end FUNCTION

REAL*4 FUNCTION CALC_A_STAT1(ALPHA,ESTPARAM, N1 ,VECTOR1, N2 , VECTOR2, M)
!Single Version
!Get the A statistic using method 1
INTEGER*2 N1, N2, M
REAL*4 ALPHA, ESTPARAM, VECTOR1(N1), VECTOR2(N2), tmp1, tmp2
tmp1 = VECTOR1(FLOOR((ALPHA)*N1))
tmp2 = VECTOR2(FLOOR((ALPHA)*N2))
CALC_A_STAT1=ABS((SQRT(FLOAT(M))*(tmp1 - ESTPARAM)) - (SQRT(FLOAT(M)/2)*
& (tmp2 - ESTPARAM)))
end FUNCTION

REAL*8 FUNCTION DCALC_A_STAT2(ALPHA,ESTPARAM, N1 ,VECTOR1, N2, VECTOR2, M)
!Double Version
!Get the A statistic using method 2
INTEGER*4 N1, N2, M
REAL*8 ALPHA, ESTPARAM, VECTOR1(N1), VECTOR2(N2), tmp1, tmp2
tmp1 = VECTOR1(FLOOR((1-ALPHA)*N1))
tmp2 = VECTOR2(FLOOR((1-ALPHA)*N2))
DCALC_A_STAT2=ABS((DSQRT(DFLOAT(M))*(tmp1 - ESTPARAM)) - (DSQRT(DFLOAT(M)/2)*
& (tmp2 - ESTPARAM)))
end FUNCTION

REAL*8 FUNCTION DCALC_A_STAT3(ALPHA,ESTPARAM, N1 ,VECTOR1, N2, VECTOR2, M)
!Double Version

```

```

!Get the A statistic using method 2
INTEGER*4 N1, N2, M
REAL*8 ALPHA, ESTPARAM, VECTOR1(N1), VECTOR2(N2), tmp1, tmp2
tmp1 = VECTOR1(FLOOR((ALPHA)*N1))
tmp2 = VECTOR2(FLOOR((ALPHA)*N2))
DCALC_A_STAT3=ABS(tmp1 - tmp2)
end FUNCTION

REAL*4 FUNCTION CALC_A_STAT2(ALPHA,ESTPARAM, N1 ,VECTOR1, N2, VECTOR2, M)
!Single Version
!Get the A statistic using method 2
INTEGER*2 N1, N2, M
REAL*4 ALPHA, ESTPARAM, VECTOR1(N1), VECTOR2(N2), tmp1, tmp2
tmp1 = VECTOR1(FLOOR((1-ALPHA)*N1))
tmp2 = VECTOR2(FLOOR((1-ALPHA)*N2))
CALC_A_STAT2=ABS((SQRT(FLOAT(M))*(tmp1 - ESTPARAM)) - (SQRT(FLOAT(M)/2)*
& (tmp2 - ESTPARAM)))
end FUNCTION

REAL*8 FUNCTION DCALC_MED( N, VECTOR )
!Double version
!get the median of a vector
USE MSIMSL
INTEGER*4 N, i
REAL*8 VECTOR(N), TEMP
CALL DSVRGN(N, VECTOR, VECTOR)
IF (MOD(N, 2) .EQ. 1) THEN
TEMP= VECTOR(CEILING(DFLOAT(N)/2))
ELSE
TEMP = (VECTOR(N/2) + VECTOR((N/2) +1))/2
ENDIF
DCALC_MED = TEMP
end FUNCTION

REAL*4 FUNCTION CALC_MED( N, VECTOR )
!Single version
!get the median of a vector

```

```

USE MSIMSL
INTEGER*4 N, i
REAL*4 VECTOR(N), TEMP
CALL SVRGN(N, VECTOR, VECTOR)
IF (MOD(N, 2) .EQ. 1) THEN
TEMP= VECTOR(CEILING(FLOAT(N)/2))
ELSE
TEMP = (VECTOR(N/2) + VECTOR((N/2) +1))/2
ENDIF
CALC_MED = TEMP
end FUNCTION

```

```

REAL*8 FUNCTION DCALC_MEAN( N, VECTOR )
!Double version
!get the mean of a vector
INTEGER*4 N, i
REAL*8 VECTOR(N)
DCALC_MEAN = sum(VECTOR)/DFLOAT(N)
end FUNCTION

```

```

REAL*4 FUNCTION CALC_MEAN( N, VECTOR )
!Single version
!get the mean of a vector
INTEGER*2 N, i
REAL*4 VECTOR(N)
CALC_MEAN = sum(VECTOR)/FLOAT(N)
end FUNCTION

```

```

REAL*8 FUNCTION DCALC_STNDERROR(B,VEC_OF_PARAMS)
!Double version
!Calculate the standard error of bootstrap replicates
USE MSIMSL
INTEGER*4 B
REAL*8 VEC_OF_PARAMS(B)
REAL*8 MEAN_OF_PARAMS
MEAN_OF_PARAMS = SUM(VEC_OF_PARAMS)/DFLOAT(B)
DCALC_STNDERROR = DSQRT(SUM((VEC_OF_PARAMS- MEAN_OF_PARAMS)**2)/(DFLOAT(B)-1))

```

```
end FUNCTION
```

```
REAL*4 FUNCTION CALC_STNDERROR(B, VEC_OF_PARAMS)
```

```
!Single version
```

```
!Calculate the standard error of bootstrap replicates
```

```
USE MSIMSL
```

```
INTEGER*2 B
```

```
REAL*4 VEC_OF_PARAMS(B)
```

```
REAL*4 MEAN_OF_PARAMS
```

```
MEAN_OF_PARAMS = SUM(VEC_OF_PARAMS)/FLOAT(B)
```

```
CALC_STNDERROR = SQRT(SUM((VEC_OF_PARAMS- MEAN_OF_PARAMS)**2)/(FLOAT(B)-1))
```

```
end FUNCTION
```

```
REAL*8 FUNCTION DCALC_SAMPLEVAR(B, VEC_OF_PARAMS)
```

```
!Double version
```

```
!Calculate the standard error of bootstrap replicates
```

```
USE MSIMSL
```

```
INTEGER*4 B
```

```
REAL*8 VEC_OF_PARAMS(B)
```

```
REAL*8 MEAN_OF_PARAMS
```

```
MEAN_OF_PARAMS = SUM(VEC_OF_PARAMS)/DFLOAT(B)
```

```
DCALC_SAMPLEVAR = (SUM((VEC_OF_PARAMS- MEAN_OF_PARAMS)**2)/(DFLOAT(B)-1))
```

```
end FUNCTION
```

```
REAL*4 FUNCTION CALC_SAMPLEVAR(B, VEC_OF_PARAMS)
```

```
!Single version
```

```
!Calculate the standard error of bootstrap replicates
```

```
USE MSIMSL
```

```
INTEGER*2 B
```

```
REAL*4 VEC_OF_PARAMS(B)
```

```
REAL*4 MEAN_OF_PARAMS
```

```
MEAN_OF_PARAMS = SUM(VEC_OF_PARAMS)/FLOAT(B)
```

```
CALC_SAMPLEVAR = (SUM((VEC_OF_PARAMS- MEAN_OF_PARAMS)**2)/(FLOAT(B)-1))
```

```
end FUNCTION
```

```
SUBROUTINE DNORMAL01VARIATES( N, OUTPUT )
```

```
!Double version
```

```

!Generate a vector of normal N(0,1) variates
USE MSIMSL
INTEGER*4 N, i
REAL*8 OUTPUT(N)
DO i =1, N
OUTPUT(i) = DRNNOF()
ENDDO
end SUBROUTINE

```

```

SUBROUTINE NORMAL01VARIATES( N, OUTPUT )
!Single version
!Generate a vector of normal N(0,1) variates
USE MSIMSL
INTEGER*2 N, i
REAL*4 OUTPUT(N)
DO i =1, N
OUTPUT(i) = DRNNOF()
ENDDO
end SUBROUTINE

```

```

SUBROUTINE DNORMALVARIATES( N,MU,SIGMA,OUTPUT )
!Double version
!Generate a vector of normal N(mu,sigma) variates
USE MSIMSL
INTEGER*4 N, i
REAL*8 OUTPUT(N), SIGMA, MU
DO i =1, N
OUTPUT(i) = (DRNNOF()*SIGMA) + MU
ENDDO
end SUBROUTINE

```

```

SUBROUTINE NORMALVARIATES( N,MU,SIGMA,OUTPUT )
!Single version
!Generate a vector of normal N(mu,sigma) variates
USE MSIMSL
INTEGER*2 N, i
REAL*4 OUTPUT(N), SIGMA, MU

```

```

DO i =1, N
OUTPUT(i) = (DRNNOF()*SIGMA) + MU
ENDDO
end SUBROUTINE

```

```

SUBROUTINE DCONTAMNORMALVARIATES ( N, MU1, SIGMA1, MU2, SIGMA2, P, OUTPUT)
!Double Version of contaminated normal
!Generates from an N(mu1, sigma1) with probability p
!Generates from an N(mu2, sigma2) with probability 1-p
USE MSIMSL
INTEGER*4 N, I
REAL*8 OUTPUT(N), SIGMA1, MU1, SIGMA2, MU2, P, UNIFORMDATA(N)
CALL DRNUN(N,UNIFORMDATA)
DO I=1,N
IF (UNIFORMDATA(I) .LE. (1-P)) THEN
OUTPUT(I) = (SIGMA1) * DRNNOF() + MU1
ELSE
OUTPUT(I) = (SIGMA2) * DRNNOF() + MU2
ENDIF
ENDDO
end SUBROUTINE

```

```

SUBROUTINE CONTAMNORMALVARIATES ( N, MU1, SIGMA1, MU2, SIGMA2, P, OUTPUT)
!Single Version of contaminated normal
!Generates from an N(mu1, sigma1) with probability p
!Generates from an N(mu2, sigma2) with probability 1-p
USE MSIMSL
INTEGER*4 N, I
REAL*4 OUTPUT(N), SIGMA1, MU1, SIGMA2, MU2, P, UNIFORMDATA(N)
CALL RNUN(N,UNIFORMDATA)
DO I=1,N
IF (UNIFORMDATA(I) .LE. (1-P)) THEN
OUTPUT(I) = (SIGMA1) * RNNOF() + MU1
ELSE
OUTPUT(I) = (SIGMA2) * RNNOF() + MU2
ENDIF
ENDDO

```

```
end SUBROUTINE
```

```
SUBROUTINE DF_VARIATES( N,DF1,DF2,OUTPUT )
```

```
!Double version
```

```
!Generate a vector of F (df1,df2) variates
```

```
USE MSIMSL
```

```
INTEGER*4 N, i
```

```
REAL*8 DF1, DF2
```

```
REAL*8 OUTPUT(N), TMP1(N),TMP2(N)
```

```
CALL DRNCHI(N,DF1,TMP1)
```

```
TMP1 = TMP1/DF1
```

```
CALL DRNGHI(N,DF2,TMP2)
```

```
TMP2 = TMP2/DF2
```

```
OUTPUT = TMP1/TMP2
```

```
end SUBROUTINE
```

```
SUBROUTINE CLRSCR()
```

```
INTEGER i
```

```
DO i=1, 25
```

```
PRINT*, ""
```

```
ENDDO
```

```
end SUBROUTINE
```

```
END MODULE
```

# Bibliography

BICKEL, P.J., GÖTZE, F. & VAN ZWET, W.R. 1997. Resampling fewer than  $n$  observations: Gains, losses, and remedies for losses. *Statistica Sinica* 7: 1-32.

BICKEL, P.J. & SAKOV, A. 2002. On the choice of  $m$  in the  $m$  out of  $n$  bootstrap in estimation problems. *Technical Report*.

CHUNG, K.H. & LEE, S.M.S. 2001. Optimal bootstrap sample size in construction of percentile confidence bounds. *Scand. J. Statist.* 28(1): 225-239.

EFRON, B. & TIBSHIRANI, R. 1993. An introduction to the bootstrap. New York: Chapman & Hall.

GÖTZE, F. & RAČKAUSKAS, A. 2001. Adaptive choice of bootstrap sample sizes. In *State of the Art in Probability and Statistics: a festschrift for W.R. van Zwet*, M. de Gunst, C. Klaassen and A. van der Vaart, eds., IMS lectures notes 36, Institute of Mathematical Statistics, Beachwood, OH, 286-309.

SWANEPOEL, J.W.H. 1986. A note on proving that the (modified) bootstrap works. *Commun. Statist.-Theory Meth.* 15(11): 3193-3203.